AD-A013 939

DEVELOPMENT OF A TRANSMISSION ERROR MODEL AND AN ERROR CONTROL MODEL

Joseph L. Hammond, Jr., et al

Georgia Institute of Technology

Prepared for:

Rome Air Development Center

May 1975

DISTRIBUTED BY:

NTIS

National Technical Information Service
U. S. DEPARTMENT OF COMMERCE

Best Available Copy

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER RADC-TR-75-138	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle) DEVELOPMENT OF A TRANSMISSION ERROR MODEL AND AN ERROR CONTROL MODEL	5. TYPE OF REPORT & PERIOD COVERED Phase Report
	6. PERFORMING ORG. REPORT NUMBER N/A
7. AUTHOR(s) J. L. Hammond J. E. Brown S. S. Liu	8. CONTRACT OR GRANT NUMBER(s) F30602-72-C-0409
9. PERFORMING ORGANIZATION NAME AND ADDRESS Georgia Institute of Technology School of Electrical Engineering Atlanta GA 30332	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 95670033
11. CONTROLLING OFFICE NAME AND ADDRESS Rome Air Development Center (RBC) Griffiss AFB NY 13441	12. REPORT DATE May 1975
	13. NUMBER OF PAGES 141
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Same	15. SECURITY CLASS (of this report) UNCLASSIFIED
	15a. DECLASSIFICATION DOWNGRADING SCHEDULE N/A

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

Same

18. SUPPLEMENTARY NOTES

RADC Project Engineer:
Woodrow W. Everett, Jr. (RBC)

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Error Control
Signal Transmission
Error Model

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The objective of this study is to choose an error detecting code for use in general purpose digital communication networks which employ automatic repeat request. To choose a code it is necessary to characterize the channels used by the communication network.

Part I of the study summarizes an extensive literature review pertaining to channel models and to design of error detecting codes. It is found that a number of channel models have been investigated and that these can be broadly

classified as renewal and nonrenewal models. Wireline and most microwave channels can be accurately represented by renewal models and model parameters have been chosen to represent practical AUTOVON channels. Nonrenewal models, which are necessary to represent for example, troposcatter channels, require more statistical parameters and are not developed to the extent of renewal models.

Part II of the report describes the development and evaluation of an algorithm for evaluating error detecting codes for use on renewal channels. The algorithm is sufficiently efficient in its use of computer time to permit an exhaustive study of possible codes with a fixed number of redundant digits.

The algorithm has been used to rank all 900 irreducible 16th degree polynomials with respect to the Pareto channel model.

For 32 check bit codes with block lengths of 2000 bits, it is shown that six classes of BCH-Fire codes encompass many of the commonly used types of codes. Three of these classes are investigated in detail in a study that considered a total of approximately 350 polynomials. There is no evidence to indicate that different results would be obtained from a study of the other three classes of BCH-Fire codes.

From this study it can be concluded that a group of possibly a dozen codes will provide the lowest undetectable error probability in general applications for which a precise channel model cannot be specified. The estimated probability of undetected errors for these "good" codes is on the order of 10^{-12}, a value which would produce one undetected error in something like fifty years at bit rates of 10^6 bits/second. Four polynomials were found to have undetected error probabilities as large as four or more orders of magnitude greater than those for good polynomials.

The code polynomial, $X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$, is recommended as specific choice. The characteristics of this polynomial are investigated in detail and it is shown that the polynomial has a probability of undetected error no larger than on the order of three times that of the best polynomial tailored to each specific channel model. For four of the channel models considered this polynomial is the best of those considered.

Part III of the report details preliminary work done in extending the results of Part II. An elementary nonrenewal Chien-Haddad model is studied. The sensitivity of the probability of undetected error to the parameters of the model and differences between pattern probabilities computed with this model and others investigated are noted.

A first step is made in developing a channel model which places in evidence the effect of physical parameters such as signal-to-noise ratio. A channel model for a DPSK modem and additive Gaussian noise is developed which, surprisingly, seems to be almost identical to models developed from practical data.

An approach to approximating nonrenewal models with renewal models is suggested. The report is concluded with recommendations for future work.

FOREWORD

The Post-Doctoral Program at Rome Air Development Center is pursued via Project 9567 under the direction of Dr. W. W. Everett, Jr. The Post-Doctoral Program is a cooperative venture between RADC and the participating universities: Syracuse University (Department of Electrical and Computer Engineering), the U. S. Air Force Academy (Department of Electrical Engineering), Cornell University (School of Electrical Engineering), Purdue University (School of Electrical Engineering), University of Kentucky (Department of Electrical Engineering), Georgia Institute of Technology (School of Electrical Engineering), Clarkson College of Technology (Department of Electrical Engineering), State University of New York at Buffalo (Department of Electrical Engineering), North Carolina State University (Department of Electrical Engineering), Florida Technological University (Department of Electrical Engineering), Florida Institute of Technology (College of Engineering), Air Force Institute of Technology (Department of Electrical Engineering), the Naval Postgraduate School (Department of Electrical Engineering), and the University of Adelaide (Department of Electrical Engineering) in South Australia. The Post-Doctoral Program provides, via contract, the opportunity for faculty and visiting faculty at the participating universities to spend a year full time on exploratory development and operational problem-solving efforts with the post-doctorals splitting their time between RADC (or the ultimate customer) and the educational institutions.

The Post-Doctoral Program is totally customer funded with projects undertaken for Air Defense Command (NORAD), Air Force Communications Service, Federal Aviation Administration, Defense Communications Agency, Aeronautical

Systems Division (AFSC), Air Force Aero Propulsion Laboratory (AFSC), Rome Air Development Center (AFSC), Electronics System Division (AFSC), NASA, Air Force Avionics Laboratory (AFSC), and Air Force Weapons Laboratory (AFSC).

This effort was undertaken for Defense Communications Agency via RADC Job Order 9567 0033 with Dr. Harry Helm and Dr. Ray Bittel as the responsible individuals at DCA-DCEC.

The authors wish to acknowl dge the assistance of Hong Wah Li and Robert Dougan during a portion of the study. The authors are with Georgia Institute of Technology: Joseph L. Hammond, Jr., is Professor of Electrical Engineering; James E. Brown is Assistant Professor Electrical Engineering (now with Atlantic Richfield); and Shyan-Shiang S. Liu, Research Assistant.

TABLE OF CONTENTS

LIST OF MAJOR SYMBOLS

$p(j)$ - probability of an error gap of length j

$F(m+1)$ - probability of an error gap greater than or equal to m+1

$a(j)$ - probability that the jth bit after an error is an error

$P(m,n)$ - probability that exactly m bit errors occur in a block of n bits

$B(m,n)$ - probability of an error burst of length m in n bits

$R(m,n)$ - probability of m-1 errors in the n-1 bits after an error

$S(m,n)$ - probability of m-1 errors in the n-1 bits after an error and the m-1 st error is in the n-1 st bit

$g(X)$ - code generator polynomial

$M(X)$ - message polynomial

$R(X)$ - check bit polynomial

$V(X)$ - code vector polynomial

$P\{\ \}$ - probability of an even

$\{0^x\ 1^y\}$ a sequence of x nonerrors followed by y errors

$\sum P_g$ - figure of merit for a code

$$S_p(n,b) - \frac{1}{n} \sum_{d_1=1}^{n-b+1} F(d_1)\, F(n - b - d_1 + 2)$$

β^* - lower bound on the probability of message patterns

β - lower bound on the probability of $P_g = \prod_{i=2}^{w} P(d_i)$

PART I

INTRODUCTION AND BACKGROUND

1. Introduction

This study is concerned with designing error detecting codes for links, of the type shown in Figure 1.1, such as might be used in future digital Defense Communications Systems.

Since the code must be designed to match the channel, the problem is two fold, namely: choosing realistic channel models and choosing good codes for specified channel models. In the most general formulation, almost any channel, line-of-sight microwave, troposcatter, wireline, or satellite, can be of interest. The codes considered have been restricted to binary linear cyclic block codes. The code should have a large block size-on the order of 2000 bits. Since the number of message bits is not to be fixed, efficient truncation of the block length should be possible. The redundancy of the code should be a multiple of 8 bit bytes with a probable choice of four such bytes for 32 bit redundancy. Finally, scrambling schemes such as NRZI should not degrade the properties of the code.

An extensive survey of the literature in the two areas of channel models and error detecting codes has been carried out. The survey reveals channel models have been studied in detail and a number of mathematical models have been matched to measured error data. The most tractable model seems to be the renewal model which is specified by the distribution function of the error gaps. Such models are good representations of line-of-sight microwave and wireline channels, while their representation for other channels is much less accurate.

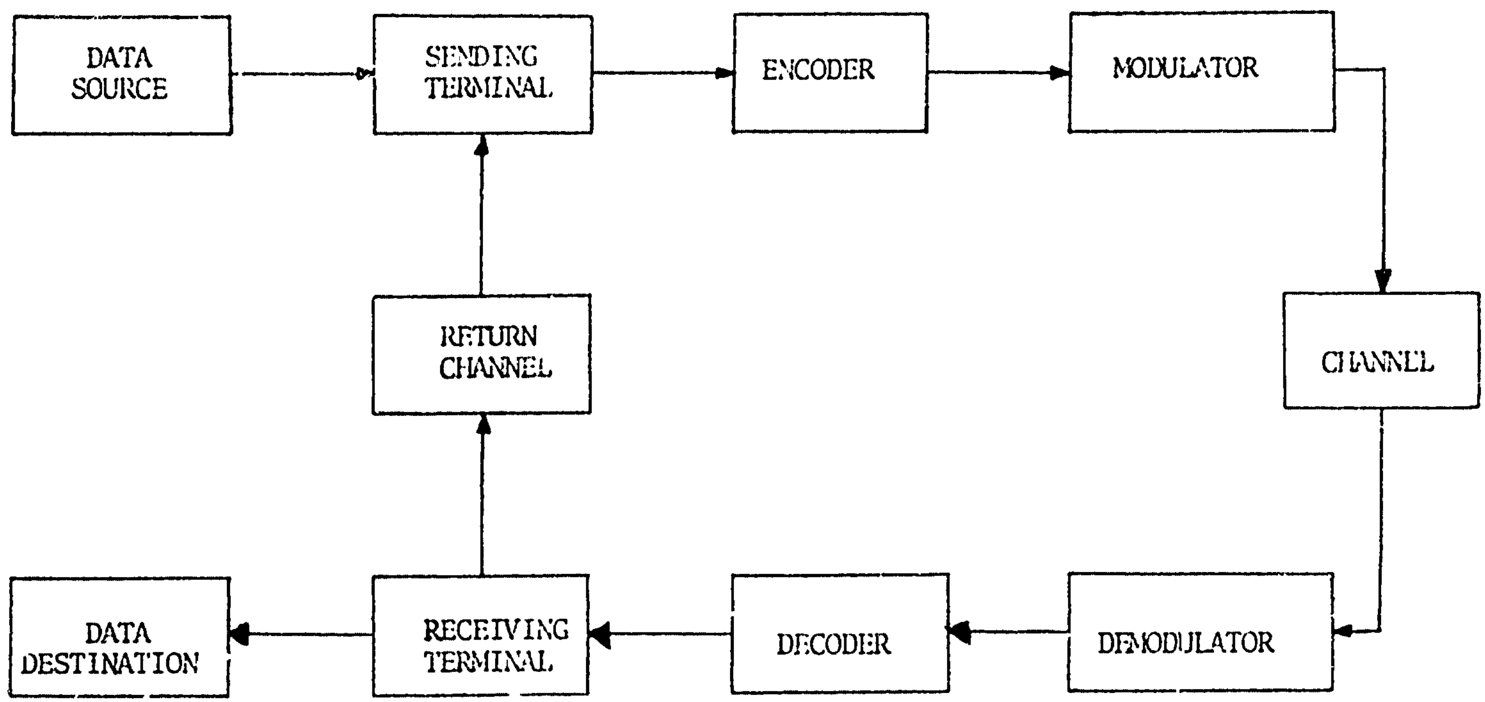


Figure 1.1 A Typical Link in A Digital Communication System

Other models, typically of a Markov type, have been used to approximate various channels. The choice of models to represent such channels as troposcatter channels remains an open question, however, for two reasons, namely: (i) there seems to be no theoretical analysis to indicate how many moments of error gap distribution are required to determine code behavior and (ii) the very large amount of experimental data required at typical error rates hampers an extensive purely empirical approach.

Given this background, it was decided to emphasize in the study the renewal type channel models which have been matched to practical channels. Thus the major portion of the contributions of the study are contained in Part II of the report on codes matched to renewal channel models.

Some preliminary work was done on more general channel models, on the problem of developing channel models based on physical parameters and on the problem of approximating nonrenewal models with renewal models. This preliminary work is presented in Part III of the report.

The remainder of Part I of the report details the review of the literature. Appendix II provides a description of the computer programs developed in the study and gives a Program Maintenance Manual.

2. Review of the Literature

The literature review is presented in three parts, namely: (a) Channel Models, (b) Matching of Models to Empirical Data and (c) Properties of Error Detecting Codes.

(a) Channel Models

In most of the work reviewed for binary communication systems, it was assumed that the message source generates a sequence $\{x_i\}$ of binary digits which are transmitted through a channel. The channel output sequence $\{y_i\}$ is a binary sequence which is the modulo-2 sum of the message sequence and an error sequence $\{e_i\}$, which is assumed to be statistically independent of the message sequence. For this structure the statistical properties of the channel are exhibited in the statistical properties of the error sequence.

A number of mathematical models are described which provide differing degrees of approximation to the measured output error patterns from typical communication equipment. The most tractable mathematical model is a renewal model which uses Pareto statistics. Most other models are Markov processes of some sort. A number of Markov processes, differing in order and definition of parameters, have been investigated. The more important general models and the references in which they are discussed are listed below:

General Model	References*†
Renewal	Elliott [1], [2]
Fritchman	Fritchman [3]
Gilbert	Gilbert [4]
Generalized Gilbert	Elliott [1], Gallager [12]
Spreading Markov	Adoul, et.al. [5]
Tsai	Fritchman [3], Tsai [6] - [8]
Chien-Haddad	Chien et.al. [9]
Pareto	Berger et.al. [10], Sussman [11]
Munter*	Munter et.al. [13]
Blank and Trafton*	Blank et.al. [14]

† References will be found at the end of Part I of this report.

* These models are not strictly Markov processes but related to Markov processes.

Efforts have been made to choose the parameters of the various models to match experimentally measured characteristics of real channels. The simplest models match the first order statistics, of such parameters as the error gaps in experimental data, to corresponding statistics of the model. More sophisticated models attempt to match higher order statistics.

The models in the above list all represent attempts to match output error statistics. Parameters in these models are not related to physical channel or modem variables. Although a considerable amount of work, such as that done by Bello [15], has been directed toward modeling analog channels in terms of their physical parameters, this work has not been carried to the point of representing digital modem output error statistics. A step in this direction, however, has been taken by Goldman [16] who computes the probability of multiple errors for a differential PSK modem for a channel represented by additive Gaussian noise and cochannel interference.

The remainder of this section defines some of the parameters necessary in discussing channel models and then presents a concise quantitative discussion of most of the models listed on page 4 .

Basic Parameters: A basic parameter for the present study is the probability, P(m,n), that exactly m bit errors occur in a transmitted block of n bits. The computation of P(m,n) is based on the statistical analysis of the number of error free bits between two bit errors. The sequence of zeroes (no errors) between the errors are called error gaps. The length of a gap is defined as one plus the total number of zeroes in the sequence between the two ones (errors). The binary error process can be equivalently described in terms of the associated gap process $\{G_i\}$, where G_i is the length of the ith gap. Define

$$\Pr\{G_n = j\} = p(j) = P(0^{j-1}\ 1|1), \tag{1.1}$$

where 0^{j-1} denotes a sequence of j-1 zeroes. The error-gap distribution

$$\begin{aligned} F(m+1) &= \sum_{j=m+1}^{\infty} p(j) \\ &= P\ (0^m|1) \end{aligned} \tag{1.2}$$

is the probability of at least m error-free bits following an error. The parameters p(j) and F(m + 1) are useful as well as P(m,n).

The autocorrelation, a(j), is the probability that the jth bit following an error is also an error; i.e.

$$a(j) = P(x^{j-1}\ 1/1) \tag{1.3}$$

where x^i denotes an arbitrary sequence of length i.

The term "error burst" plays a useful role in error analysis even though no generally accepted definition seems to exist. Intuitively an error burst is identified as a sequence beginning and ending with an error with relatively large gaps in either side of it compared to the gaps within the burst. The notation B(m,n) will be used to designate the probability of an error burst of length m in a sequence of n bits.

The probability that m - 1 errors occur in the n - 1 bits following an error is denoted R(m,n). A related statistic of interest is the probability of m - 1 errors in the n - 1 bits following an error with the (m - 1)th error in the (n - 1)st bit. The notation S(m,n) will be used for this statistic.

The more important of these basic parameters are evaluated for certain specific models and are plotted in Appendix I.

Renewal Channels: For a renewal channel (lengths of gaps independent) the probability of error patterns are easily computed. Let $\beta(d_1, d_2, \ldots, d_{m+1}) = \beta(\vec{d})$ correspond to an error pattern consisting of n consecutive bits containing m errors where there are d_i zeroes before the ith error and d_{m+1} zeroes after the last error. The probability of this pattern is expressed as

$$P\left[\beta(\vec{d})\right] = P(0^{d_1}\,1) \cdot \prod_{i=2}^{m} P(0^{d_i}\,1\,|\,1) \cdot P(0^{d_{m+1}}\,|\,1) \tag{1.4}$$

$$= P(1)F(d_1 + 1) \prod_{i=2}^{m} p(d_i + 1)\, F(d_{m+1} + 1)\,,$$

Elliot [2] proceeds to establish

$$P(m,n) = \sum_{j=1}^{n-m+1} P(1)\, F(j)\, R(m,n-j+1),\quad 1 \le m \le n, \tag{1.5}$$

and

$$R(m,n) = \begin{cases} F(n) & m = 1,\ n \ge 1 \\ \sum_{j=1}^{n-m+1} p(j)\, R(m-1,n-j) & 2 \le m \le n\ ,\quad n \ge 2\,. \end{cases} \tag{1.6}$$

Alternatively, for renewal channels, the autocorrelation, a(j), of the bit errors can be used to specify the channel. Elliott [2] determines a(j) by the recursion

$$a(j) = \begin{cases} 1 & j=0 \\ p(1) & j=1 \\ p(j) + \sum_{s=1}^{j-1} p(s)a(j-s) & j > 1\ . \end{cases} \tag{1.7}$$

The Binary Symmetric Channel Model: The simplest renewal channel is the binary symmetric channel. The channel is a memoryless channel with the probability of either type of error being given by q. It is straightforward to establish that

$$P(j) = q(1 - q)^{j-1} \quad , \tag{1.8}$$

$$F(m + 1) = (1 - q)^{m} \quad , \tag{1.9}$$

and

$$P(m,n) = \binom{n}{m} q^{m} (1 - q)^{n-m} \tag{1.10}$$

This channel is almost trivial to analyze. Unfortunately, it is seldom applicable to a physical communication system.

Pareto Model [10]: Berger and Mandelbrot proposed a renewal model with the error gap distribution given by the Pareto distribution

$$F(m) = 1/m^{\delta}, \tag{1.11}$$

where δ is a positive constant less than 1. Since

$$\sum_{m} F(m) = \infty \quad , \tag{1.12}$$

the channel model does not have finite recurrence times; i.e. the average number of symbols between two errors is infinite. This problem is resolved by letting δ take on a new constant value greater than unity at some value $m = m^*$. The value of δ for $m < m^*$ and the value of m^* are the parameters of the model.

The more common application of the model is to consider two truncation parameters, m_* and m^*. In this study, the error gap distribution was chosen to be of the form

$$F(m) = \begin{cases} 1 & , m = 0 \\ \left(\dfrac{L^\alpha}{L^\alpha - 1}\right)\left[\dfrac{1}{m^\alpha} - \dfrac{1}{L^\alpha}\right], & 1 \le m \le L \\ 0 & , m > L \end{cases} \tag{1.13}$$

where

$$L = \left[\frac{1-\alpha}{\alpha} \quad (E(n) + 1)\right]^{\frac{1}{1-\alpha}} \tag{1.14}$$

and E(n) is the average gap length given by

$$E(n) = \frac{\alpha}{1-\alpha} \quad L^{1-\alpha} - 1. \tag{1.15}$$

Gilbert Channel Model [4]: The channel model proposed by Gilbert consists of a two-state first-order Markov chain composed of a good state C_1 and a bad state C_2. The good state is error free; the bad state has error probability δ. The state transitions occur synchronously with the transmission of the input symbols according to the state transition probabilities

$$t_{ij} = P\ (C_i \rightarrow C_j)\ . \tag{1.16}$$

The process is assumed to be stationary.

The Gilbert model can be transformed into a three-state first-order Markov chain composed of two error-free states C_1' and C_2' and an error state C_3' with the transition matrix

$$T' = \begin{bmatrix} t_{11} & (1-\delta)t_{12} & \delta t_{12} \\ t_{21} & (1-\delta)t_{22} & \delta t_{22} \\ t_{21} & (1-\delta)t_{22} & \delta t_{22} \end{bmatrix} \tag{1.17}$$

A generalization of this channel model is the Fritchman model discussed below.

Fritchman Channel Model [3]: The channel model proposed by Fritchman consists of an N-state Markov chain whose state space is partioned into two groups of states. The first K states are error-free and the last N-K states are error states. The state transitions occur synchronously with the transmission of the input symbols according to the state transition probabilities

$$t_{ij} = P\left(C_i \rightarrow C_j\right) . \tag{1.18}$$

The process is assumed to be stationary.

The error process $\{e_t\}$ is generated as follows: Partition the N states into the two subsets

$$A = \{C_1, C_2, \ldots, C_k\} \tag{1.19}$$

and

$$B = \{C_{k+1}, \ldots, C_N\} \tag{1.20}$$

Let $\{z_t\}$ denote the state process. Define

$$\phi\,(C_i) = \begin{cases} 0 & C_i \in A \\ 1 & C_i \in B \end{cases} \tag{1.21}$$

The process is defined by

$$e_t = \phi(z_t) . \tag{1.22}$$

The transition matrix $T_A = \{t_{ij}^A\}$ among the A states is assumed to be similar to a diagonal matrix with

$$L^{(i)} = \left(\ell_1^{(i)}, \ldots, \ell_K^{(i)}\right), \quad 1 \le i \le K , \tag{1.23}$$

and

$$R^{(i)} = \left(r_i^{(i)}, \ldots, r_K^{(i)}\right), \quad 1 \le i \le K , \tag{1.24}$$

corresponding to the left and right eigenvectors of T_A for the eigenvalue λ_i. The m-step transition probabilities may be expressed as

$$t_{ij}(m) = \sum_{k=1}^{N} a_k \, r_i^{(k)} \, \ell_j^{(k)} \, \lambda_k^m \tag{1.25}$$

where

$$a_k = \left[\sum_{i=1}^{K} r_i^{(k)} \, \ell_i^{(k)}\right]^{-1}$$

Fritchman proceeds to establish that the error-gap distribution is given by

$$F(m+1) = \sum_{i=1}^{K} f_m(i) \, \lambda_i^m \tag{1.26}$$

where

$$f_m(i) = \begin{cases} \dfrac{1}{\lambda_i} \quad \dfrac{\sum_{j=K+1}^{N} \sum_{\ell=1}^{K} \mu_j t_{j\ell}}{\sum_{j=K+1}^{N} \mu_j} & m = 1 \\[2ex] \dfrac{a_i}{\lambda_i} \quad \dfrac{\sum_{j=K+L}^{N} \sum_{\ell=1}^{K} \sum_{m=1}^{K} \mu_j t_{j\ell} r_\ell^{(i)} \ell_m^{(i)}}{\sum_{j=k+1}^{N} \mu_j} & m \gtrsim 2 \end{cases} \tag{1.27}$$

The μ_j correspond to the steady-state probabilities of the channel states c_j.

If the transition matrix $T_B = (t_{ij}^B)$ among the B states is assumed to be similar to a diagonal matrix with

$$L^{(i)} = \left(\ell_{K+1}^{(i)}, \ldots, \ell_N^{(i)}\right), \quad K+1 \le i \le N, \tag{1.28}$$

and

$$R^{(i)} = \left(r_{K+1}^{(i)}, \ldots, r_N^{(i)}\right), \quad K+1 \le i \le N, \tag{1.29}$$

corresponding to the left and right eigenvectors of T_B for the eigenvalue λ_i, then the error-cluster distribution may be expressed as

$$P\left(1^m|0\right) = \sum_{i=K+1}^{N} f_m(i)\ \lambda_i^{m-1}, \tag{1.30}$$

where

$$f_m(i) = \left(\frac{a_i}{\lambda_i}\right) \frac{\sum_{j=1}^{K} \sum_{\ell=K+1}^{N} \sum_{m=K+1}^{N} \mu_j t_{j\ell}\ r_\ell^{(i)} \ell_m^{(i)}}{\sum_{j=1}^{K} \mu_j} \tag{1.31}$$

and

$$a_k = \left[\sum_{i=K+1}^{N} r_i^{(k)} \ell_i^{(k)}\right]^{-1}. \tag{1.32}$$

Tsai Channel Model: Fritchman [31] identifies a special case of his general model consisting of $K = N-1$ error-free states and a single error state. This model, which was later studied in detail by Tsai [6], [7], has a transition matrix given by

$$T = \begin{bmatrix} t_{11} & 0 & \cdots & t_{1N} \\ 0 & t_{22} & \cdots & t_{2N} \\ \cdot & \cdot & \ddots & \cdot \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & t_{N-1,N-1} & \cdot \\ t_{N1} & t_{N2} & \cdots & t_{N,N} \end{bmatrix} \tag{1.33}$$

Note that there are no transitions between the error-free states. The state transitions occur synchronously with the transmitted bits.

It follow directly from Fritchman's model that the error-gap distribution is given by

$$F(m+1) = \sum_{k=1}^{N-1} t_{Nk} \left(t_{kk}\right)^{m-1}, \quad m \geq 1. \tag{1.34}$$

The error-gap mass density function is given by

$$p(j) = F(j) - F(j+1)$$

$$= \begin{cases} t_{NN} & j = 1 \\ \sum_{k=1}^{N-1} t_{Nk}\, t_{kk}^{j-2}\, t_{kN} & j \geq 2 \end{cases} \tag{1.35}$$

Tsai uses an error burst defined by Brayer [24] as a sequence:

1. beginning and ending with an error,

2. the ratio of the number of errors to the number of digits larger than or equal to a specified number δ,

3. if the inclusion of the next error keeps the ratio above the specified number δ, the burst is continued; otherwise the burst ends, and

4. not beginning with an error belonging to the previous burst.

A burst interval is the region between two bursts. Obviously, the length of an error burst and the burst interval will be affected by the choice of δ.

The probability of a burst of length m with n errors, $B_n(m)$, is calculated as following: Let S(n,m) be the probability of a sequence of m digits with n errors satisfying $m_1 = 1$, $m_n = m$, and

$$\frac{i}{m_i} \geq \delta, \quad 1 \leq i \leq n ,$$

where m_i is the length up to the ith error. It follows that

$$S(n,m) = \begin{cases} p(m-1) & n = 2,\ m \geq 2 \\ \\ \displaystyle\sum_{i=n-1}^{\min\left\{\frac{n-1}{\delta},\ m-1\right\}} S(n-1,i)\, p(m-i) & n > 2,\ 2 \leq m \leq 2/\delta \end{cases} \tag{1.36}$$

where fractions are to be taken as the largest integer less than the fraction. Thus,

$$B_n(m) = S(n,m)\ \Pr\left\{m_{n+1} > (n+1)/\delta\right\} \tag{1.37}$$

where the fraction is to be taken as the integer greater than the fraction. Noting that

$$\Pr\left\{m_{n+1} > (n+1)/\delta\right\} = \Pr\left\{m_{n+1} - m > (n+1)/\delta - m\right\}$$
$$= F\left[(n+1)/\delta - m\right] \tag{1.38}$$

one concludes

$$B_n(m) = S(n,m)\ F\left[(n+1)\ \delta - m\right]\ ,\quad n \geq 2. \tag{1.39}$$

For n = 1, a burst consists of a single error. Hence,

$$B_1(1) = F\left[(n+1)\delta - 1\right]\ ,\quad n = 1. \tag{1.40}$$

The probability, B(m), of a burst of length m is given by

$$B(m) = \sum_{n=m\delta}^{m} B_n(m)$$
$$= \sum_{n=m\delta}^{m} S(n,m)\ F\left[(n+1)\ \delta - 1\right] \tag{1.41}$$

since $n/m_n \geq \delta$ by definition of burst.

Slowly Spreading Markov Chain Model [5]: This channel model, suggested by Adoul, is an extension of the Fritchman model to a denumerably infinite-state Markov chain (slowly spreading Markov chain). Let $\{z_n\}$ denote the state process. The error sequence $\{e_n\}$ is defined by

$$e_n = \begin{cases} 1 & z_n = 0 \\ 0 & z_n \neq 0 \end{cases} \tag{1.42}$$

The state-transition probabilities are given by

$$t_{ij} = \begin{cases} q_i & j = 0 \\ p_i & j = i+1 \\ 0 & \text{otherwise} \end{cases} \tag{1.43}$$

The state transitions are assumed to occur synchronously with the transmitted symbols.

It is obvious that for this mod·ˡ, the error-gap distribution is given by

$$F(m + 1) = \prod_{k=1}^{m} p_k \tag{1.44}$$

and the error-gap mass density function is given by

$$p(j) = q_j \prod_{k=1}^{j-1} p_k. \tag{1.45}$$

This model allows a very general specification of a renewal process. The only constraint is that the error state must be recurrent; i.e. the probability of eventually returning is unity. The return to the error state can take a very large number of transitions. The expected or average number of states for the first return to the error state is

$$\begin{aligned} E[G_n] &= \sum_{j=1}^{\infty} j\, p(j) \\ &= \sum_{j=1}^{\infty} j \left[F(j) - F(j+1)\right] \\ &= \sum_{j=1}^{\infty} F(j) \end{aligned} \tag{1.46}$$

Hence, the specification is arbitrary up to the constraint

$$\sum_{j=1}^{\infty} F(j) < \infty .$$

Munter and Wolf Channel Model [13]: The model proposed by Munter and Wolf consists of combining M renewal channels in such a manner that the resulting composite channel is not itself a renewal process. Specifically, the error bits occurring in a time interval $\lfloor n_o + \ell N, n_o + (\ell+1) N - 1 \rfloor$ are generated by a renewal process (channel) C_i with probability λ_i. At time $n_o + (\ell+1) N$ a new renewal process C_j is chosen with probability λ_j, independently of the previous renewal processes. The error bits occurring in the time interval $\lfloor n_o + (\ell+1) N, n_o + (\ell+2) N - 1 \rfloor$ are generated by the renewal process C_j. In general, a new renewal process is selected every N samples, independently of the previous choices. The starting time n_o is equally likely to be 0, 1, ..., N - 1.

The autocorrelation, a(j), of the errors is given by

$$a(j) = \frac{\sum_{i=1}^{M} \lambda_i \, P_i(1) \left[\frac{N-j}{N} \, a_i(j) + \frac{j}{N} \sum_{\ell=1}^{M} \lambda_\ell \, P_\ell(1) \right]}{\sum_{i=1}^{M} \lambda_i \, P_i(1)} , \quad 0 \le j \le N, \tag{1.47}$$

where

$$a_i(j) = \text{error autocorrelation for } C_i$$

and

$$P_i(1) = \text{probability of bit error for } C_i .$$

The derivation is based on the assumption that at each channel selection time, $n_o + \ell N$, a new error sequence begins independent of the preceeding error sequences; i.e. even if the same renewal process remains in effect, the new error sequence is independent of the previous one. This assumption can be relaxed such that if the same process remains in effect, the new error sequence is a continuation of the preceeding process. The resulting autocorrelation is

$$a(j) = \frac{\sum_{i=1}^{M} \lambda_i P_i(1) \left[\left(\frac{N-j}{N} + \frac{j}{T} a_i(j) + \frac{j}{N} \sum_{\substack{\ell=1 \\ \ell \neq i}}^{M} \lambda_\ell P_\ell(1) \right) \right]}{\sum_{i=1}^{M} \lambda_i P_i(1)}, \quad 0 \le j \le N. \tag{1.48}$$

Blank and Trafton Channel Model [14]: Blank and Trafton consider a generalization of Elliott's renewal channel model for which the error process is characterized by an n-state m-th order Markov error-state model with each error state consisting of a renewal error process. The state of the channel is allowed to change only when an error occurs. The renewal processess are re-initialized at that time. The composite channel is non-renewal, in general. An analysis of this model is given in the reference cited.

Generalized Gilbert Channel Model, [1]: The generalized Gilbert channel model consists of a two-state first-order Markov chain. Each state (channel) is characterized as a binary symmetric channel with error probability q_i, i = 1, 2. The state transitions occur synchronously with the transmitted bits. The state transition matrix is given by $T = (t_{ij})$, where t_{ij} denotes the probability of moving to state C_j from C_i.

The characteristics of this model may be obtained from the analysis of the Chien-Haddad model which is a generalization of this model.

Chien-Haddad Model [9]: The channel model proposed by Chien, et. al. consists of an N state first-order Markov process. Corresponding to each state C_i, the channel is characterized by a binary symmetric channel with error probability q_i. The state-transition probabilities are given by

$$t_{ij} = \Pr\{C_i \rightarrow C_j\}$$

with the transitions occuring synchronously with the transmitted symbols. The steady-state probabilities $\{\pi_i\}$ are given as the elements of the vector π satisfying

$$\pi T = \pi .$$

To establish the error-gap distribution proceed as follows:

Note that

$$F(m+1) = P(0^m|1)$$

$$= \sum_k \sum_\ell \Pr\{0^m, \text{ last state } C_\ell \mid \text{one in state } C_k\} P\{C_k|1\} \tag{1.49}$$

$$= \frac{1}{P(1)} \sum_k \sum_\ell P_r \{0^m, \text{ last state } C_\ell | x \text{ in state } C_k\} \pi_k q_k$$

where

$$P(1) = \sum_{i=1}^{N} \pi_i q_i \quad . \tag{1.50}$$

Define

$$Q_{k\ell}(m) = \Pr\{0^m, \text{ last state } C_\ell | x \text{ in state } C_k\} . \tag{1.51}$$

Note that

$$Q_{k\ell}(m) = \sum_j t_{kj} (1-q_j) Q_{j\ell}(m-1) \tag{1.52}$$

In matrix notation

$$Q(m) = DQ(m - 1) \qquad (1.53)$$

with $Q(0) = I$, where

$$Q(m) = \{Q_{ij}(m)\},$$

$$D = T[I - \Delta(q)],$$

$$q = (q_1, \ldots, q_N) .$$

and $\Delta(q)$ corresponds to the diagonal matrix whose diagonal elements are the elements of the vector q. Hence,

$$Q(m) = D^m \qquad (1.54)$$

and

$$F(m+1) = \frac{1}{P(1)} \sum_{k=1}^{N} \sum_{\ell=1}^{N} \pi_k q_k Q_{k\ell}(m) \qquad (1.55)$$

$$= \frac{1}{P(1)} \pi \Delta(q) D^m e' ,$$

where e' denotes the transpose of the vector

$$e = (1, \ldots, 1) .$$

In terms of the eigenvalues λ_i of D,

$$F(m+1) = \frac{1}{P(1)} \sum_{i=1}^{N} a_i \lambda_i^m , \qquad (1.56)$$

where

$$a_i = \pi \Delta(q) B(i) e' ,$$

and

$$\left[I - zD\right]^{-1} = \sum_{i=1}^{N} B(i)\left(1 - \lambda_i z\right)^{-1} \quad . \tag{1.57}$$

Note that the form of the error-gap distribution is equivalent to the Tsai channel model. However, the determination of the parameters $\{a_i\}$ and $\{\lambda_i\}$ are not sufficient to uniquely characterize this model since it does not correspond to a renewal model. The Tsai channel model can be obtained as a special case of this model if the only non-zero elements of T are t_{ii}, t_{iM}, and t_{Mi} together with $q_i = 0$, $1 \le i \le N - 1$, and $q_N = 1$.

The unique characterization of the Chien-Haddad channel model depends on determining the higher order statistics of the gap process $\{G_n\}$. Designate

$$F(m+1;\ n+1) = \Pr\left\{G_{i+1} \ge m+1,\ G_i = n+1\right\} \quad . \tag{1.58}$$

A similar derivation to the preceeding one yields

$$\begin{aligned} F(m+1;\ n+1) &= \frac{1}{P(1)}\,\pi\Delta(q)D^{n}T\Delta(q)D^{m}e'. \\ &= \frac{1}{P(1)}\sum_{i=1}^{N}\sum_{j=1}^{N} a_{ij}\,\lambda_i^{n}\,\lambda_j^{m} \end{aligned} \tag{1.59}$$

where

$$a_{ij} = \pi\Delta(q)B(i)T\Delta(q)B(j)e' \ .$$

The conditional error-gap distribution is given by

$$\begin{aligned} F(m+1\,|\,n+1) &= \frac{F(m+1;n+1)}{F(n+1)-F(n+2)} \\ &= \frac{\sum_{i=1}^{N}\sum_{j=1}^{N} a_{ij}\lambda_i^{n}\lambda_j^{m}}{\sum_{i=1}^{N} a_i\lambda_i^{n}(1-\lambda_i)} \end{aligned}$$

$$= \sum_{j=1}^{N} a_j(n) \lambda_j^m \quad .$$

where

$$a_j(n) = \frac{\sum_i a_{ij} \lambda_i^n}{\sum_i a_i (1-\lambda_i) \lambda_i^n} \quad .$$

The quantity P(m, n) can be computed for the Chien-Haddad model using a recursion relation which is now given. The probability P(m, n) is given by

$$P(m, n) = \sum_i P_i(m, n)$$

where the sum extends over the states, C_i, of the model and

$$P_i(m, n) = P\{m \text{ errors in a block of length } n, \text{ last bit is from } C_i\}.$$

The quantity $P_i(m, n)$ is then expressed as

$$P_i(m, n) = \sum_{j=1}^{N} [P_j(m-1, n-1)\, t_{ji}\, q_i + P_j(m, n-1)\, t_{ji}\, (1-q_i)] \tag{1.61}$$

with the initial condition

$$P_j(0, 1) = (1 - q_j)\, \pi_j$$

$$P_j(1, 1) = q_j\, \pi_j \; ,$$

The result can be expressed in matrix form by defining a vector $\underline{P}(m,n)$ as

$$\begin{aligned}\underline{P}(m,n) &= [P_1(m,n),\ P_2(m,n),\ \ldots,\ P_N(m,n)] \\ &= \underline{P}(m-1,\ n-1)\ T\Delta(q) + \underline{P}(m,n-1)\ T[I - \Delta(q)]\end{aligned} \tag{1.62}$$

The probability P(m,n) is then given by

$$P(m,n) = \underline{P}(m,n)\ e' \tag{1.63}$$

where

$$\underline{P}(0,n) = \pi D^n \ , \quad n \geq 0.$$

The result for computing B(b, N)† is given in matrix notation as

$$B(b,N) = \pi \sum_{d=0}^{N-b} D^d\ R(b)\ D^{N-b-d}\ e' \tag{1.64}$$

where

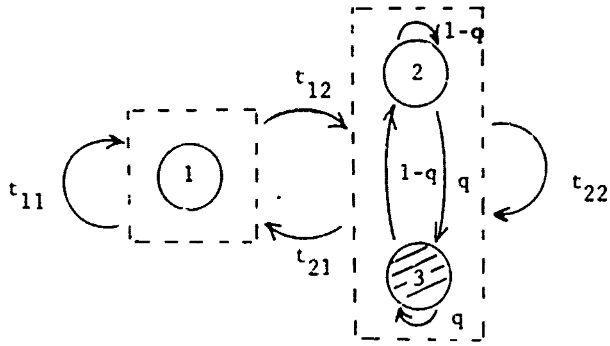
$$R(b) = T\Delta(q)\ T^{b-1}\ \Delta(q)$$

$$B(0,\ N) = \pi\ D^N\ e' \ .$$

A useful summary of the channel models is given by the state transition diagrams of Figure 1.2 for renewal models and Figure 1.3 for nonrenewal models.

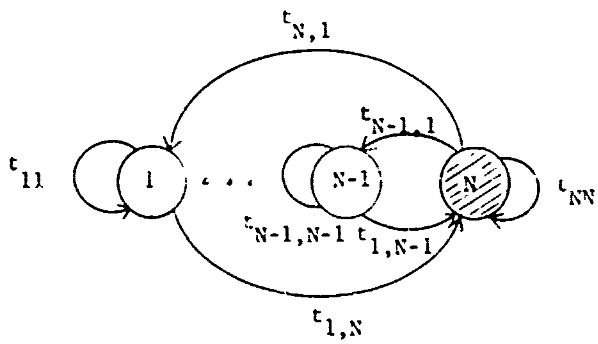
---

†An "error burst" is defined here as starting with an error and ending with an error.



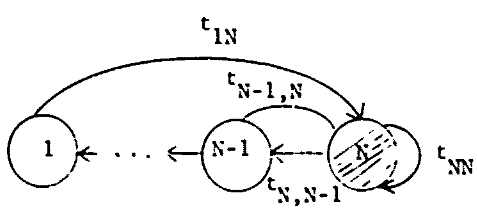
$$T = \begin{bmatrix} t_{11} & (1-q)t_{12} & qt_{12} \\ t_{21} & (1-q)t_{22} & qt_{22} \\ t_{21} & (1-q)t_{22} & qt_{22} \end{bmatrix}$$

Gilbert Model



$$T = \begin{bmatrix} t_{11} & 0 & \cdots & t_{1N} \\ 0 & t_{22} & \cdots & t_{2N} \\ \vdots & & \ddots & \vdots \\ t_{N1} & t_{N2} & \cdots & t_{NN} \end{bmatrix}$$

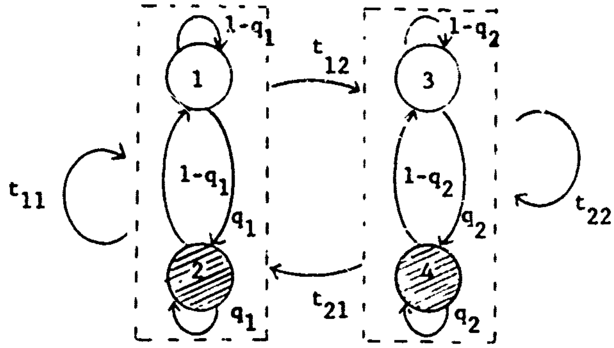
Tsai Model (Special Case of Fritchman Model)



$$T = \begin{bmatrix} 0 & 0 & \cdots\cdots\cdots & t_{1N} \\ t_{21} & 0 & \cdots\cdots\cdots & t_{2N} \\ 0 & t_{31} & \cdots\cdots\cdots & t_{3N} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots \; t_{N-1,N} & t_{NN} \end{bmatrix}$$

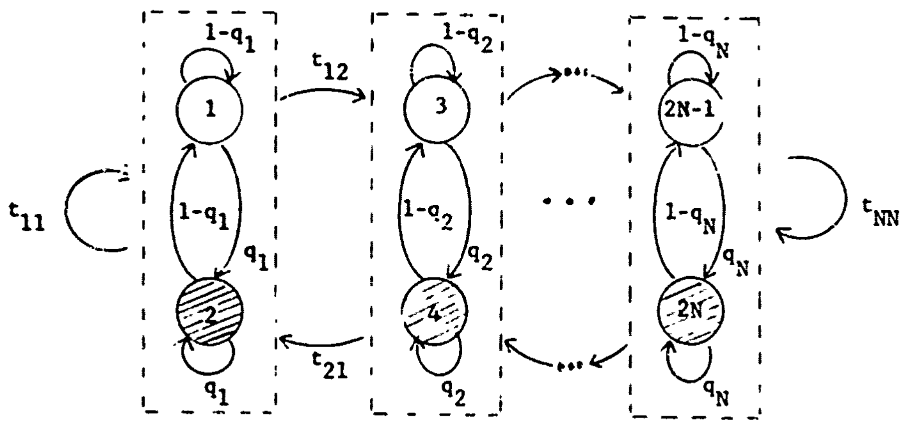
Spreading Markov Model

Figure 1.2. Special Cases of Renewal Models

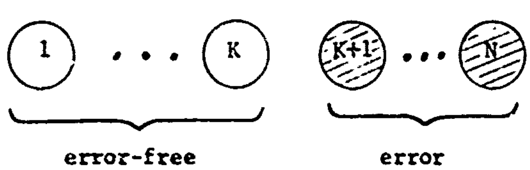


$$T = \begin{bmatrix} (1-q_1)t_{11} & q_1t_{11} & (1-q_2)t_{12} & q_2t_{12} \\ (1-q_1)t_{11} & q_1t_{11} & (1-q_2)t_{12} & q_2t_{12} \\ (1-q_1)t_{21} & q_1t_{21} & (1-q_2)t_{22} & q_2t_{22} \\ (1-q_1)t_{21} & q_1t_{21} & (1-q_2)t_{22} & q_2t_{22} \end{bmatrix}$$

Generalized Gilbert Model



Chien-Haddad Model



Fritchman Model

Figure 1.3. Special Cases of Nonrenewal Models

b. Matching of Channel Models to Experimental Data

Most of the authors listed in the previous section have made an attempt to match their channel models to empiracal data taken from real channels. For example, Elliott [1], [2], Gilbert [4] and Munter and Wolf [13] work with data for switched telephone networks such as presented by Townsend and Watts [22]. Tsai [7], [8], and Fritchman [3] use data for HF channels and Chien et.al. [9] and Tsai [6] treat troposcatter channels.

Possibly the most thorough study concerned with matching channel models to real channel data has been conducted by Brayer [23], [24], [25], [26] who considers HF, troposcatter, satellite and wireline channels. Extensive empirical data for troposcatter channels is analyzed by Chien et.al. [27].

As a concise summary of the literature, it can be stated that wireline and HF channels have the characteristics of renewal models and hence can be modeled with good accuracy. Troposcatter channels are definitely not renewal in nature. For these channels the modeling problem is much more complicated and the choice of good models seems to be still an open question. The remainder of this section will discuss techniques for matching channel models to experimental data.

Renewal channel models have the tractable property that a first order statistic such as the error gap distribution, $F(m + 1)$, completely defines the model. For many renewal channels, the model parameters $\{t_{ij}\}$ can be obtained by fitting the function

$$\sum_{k=1}^{N-1} a_k \lambda_k^m$$

to the experimentally measured error-gap distribution F(m+1). The model parameters are found from

$$t_{kk} = 1 - t_{kN} = \lambda_k \ , \ 1 \le k \le N-1 \ , \qquad (1.65)$$

$$t_{Nk} = a_k \lambda_k \ , \ 1 \le k \le N-1 \ ,$$

and

$$t_{NN} = 1 - \sum_{k=1}^{N-1} t_{Nk} \ . \qquad (1.66)$$

The Munter and Wolf model [13] effectively consists of M renewal channels C_1, C_2 ... C_i, ... C_n and hence represents a more complicated channel than the renewal one. If the model is applied to codes with fixed block lengths much less than N, (recall that N determines the time spacing of the renewal process), and the component channels have the same error rate, the error autocorrelation may be approximated by

$$a(j) \approx \sum_{i=1}^{M} \lambda_i \ a_i(j) \qquad j \ll N \qquad (1.67)$$

for both channel models.

It is also possible to establish that

$$p(j) \approx \sum_{i=1}^{M} \lambda_i \ p_i(j) \ , \qquad j \ll N \qquad (1.68)$$

and

$$P(m,n) \approx \sum_{i=1}^{M} \lambda_i \ P_i(m,n) \ , \quad m(n-1) \ll N \ , \qquad (1.69)$$

where

$$P_i(j) = \text{the error-gap mass-density function for } C_i$$

and

$$P_i(m,n) = \text{the probability that exactly m bit errors occur in a transmitted block of n bits for } C_i.$$

The particular class of renewal channel used in these formula will depend upon the error data. Munter and Wolf [6] consider the Gilbert [9] renewal channel model in which

$$a_i(j) = \alpha_i K_i^j + P_i\,(1)\;, \quad j \geq 1\;. \tag{1.70}$$

Assuming that

$$P_i(1) << \alpha_i\, k_i^j\;, \qquad 1 \leq j << N\;,$$

and

$$\frac{\alpha_i\, k_i}{1-k_i(1-\alpha_i)} \approx 1\;,$$

it is shown that*

$$P_i(m,n) = P_i(1)\,\frac{\binom{n}{m}\alpha_i^{m+1}\,k_i^{n+1}(1-\alpha_i)^{n-m}}{[1-k_i(1-\alpha_i)]^2}\;, \quad 1 \leq m \leq n. \tag{1.71}$$

The application of this model to actual data consists of the following steps:

---

*There is a possible inconsistency in (1.69) and (1.71). See Appendix III for a discussion of this point.

1. Plot the experimentally measured error data $P(m,n)/\binom{n}{m}$ as a function of m for various values of n.

2. Approximate each curve by straightline segments parallel to one another for different values of n.

3. From the theoretical model, one has

$$P(m,n)\Big/\binom{n}{m} = \sum_{i=1}^{M} \lambda_i \; P_i(1) \; \frac{\alpha_i^{m+1} \; K_i^{n+1} \; (1-\alpha_i)^{n-m}}{[1 - K_i(1-\alpha_i)]^2}$$

Let $P_i(1)$ be the average error rate of the data. The ith set of straightline approximations are matched to the ith term in the summation. The slope of the ith approximation is

$$\log \left[\alpha_i \;/\; (1-\alpha_i)\right] ;$$

the vertical separation between the ith approximations resulting from changing n by Δn is

$$\Delta n \;\; \log \left[K_i(1-\alpha_i)\right] ;$$

and the vertical positioning of the ith segments is specified by λ_i.

The Chien-Haddad model, which is one of the most general reviewed in this report, requires both first and second order statistics of the error process. Consider the problem of determining the model based on knowledge of P(1), F(m+1), and F(m+1; n+1) as defined in (1.58). Restrict attention to the case for which D is similar to a diagonal matrix; i.e.

$$D = M\Delta(\lambda)M^{-1}. \tag{1.72}$$

Note that

$$[I - zD]^{-1} = M[I - z\Delta(\lambda)]^{-1} M^{-1}$$

$$= \sum_{i=1}^{N} r_i' u_i (1-\lambda_i z)^{-1} , \qquad (1.73)$$

where

$$M = \left[r_1' \,\vdots\, r_2' \,\vdots\, \cdots \,\vdots\, r_N' \right] ,$$

r_i is the right eigenvector of D corresponding to λ_i ,

$$M^{-1} = \begin{bmatrix} u_1 \\ \hline u_2 \\ \hline \vdots \\ \hline u_N \end{bmatrix} ,$$

u_i is the left eigenvector of D corresponding to λ_i ,

and

$$u_i r_j' = \delta_{ij} .$$

Hence,

$$B(i) = r_i' u_i . \qquad (1.74)$$

For convenience, normalize the eigenvectors $\{r_i'\}$ such that†

$$Me' = \sum_{i=1}^{N} r_i' = e' . \tag{1.75}$$

Note that

$$e' = M^{-1}Me' = M^{-1}e' .$$

One, therefore, has

$$F(m+1) = \frac{1}{P(1)} \sum_{i=1}^{N} a_i \lambda_i^m \tag{1.76}$$

where

$$a_i = \pi\Delta(q)\, r_i' .$$

Define the vector

$$a = (a_1, \ldots, a_N) . \tag{1.77}$$

Note that

$$\begin{aligned} a &= \pi\Delta(q)M \\ &= \pi T\Delta(q)M \\ &= \pi[T - M\Delta(\lambda)M^{-1}]M \\ &= \pi M - \pi M\Delta(\lambda) . \end{aligned}$$

†The fact that this can be done is based on observing that the eigenvectors may be expressed as $r_i' = c_i e_i'$, where $e_i e_i' = 1$ and c_i is an arbitrary constant. Hence, $Me' = \sum_{i=1}^{N} c_i e_i'$. Moreover, the e_i' forms a basis. Hence $e' = \sum_{i=1}^{N} d_i e_i'$. Therefore, choosing $c_i = d_i$ results in the appropriate normalization.

Similarly, for the joint error-gap distribution one obtains

$$a_{ij} = \pi \Delta(q)\ r_i'\ u_i\ T\ \Delta(q) r_j'$$

$$= a_i\ u_i T\Delta(q) r_j' \ . \tag{1.78}$$

Define the matrix

$$A = \{a_{ij}\}$$

$$= \Delta(a)\ M^{-1}\ T\Delta(q)M \tag{1.79}$$

Observing that

$$T = D[I - \Delta(q)]^{-1}$$

$$= M\Delta(\lambda)M^{-1}\ [I - \Delta(q)]^{-1}$$

enables one to express

$$A = \Delta(a)\ \Delta(\lambda)M^{-1}\ [I - \Delta(q)]^{-1}\ \Delta(q)M$$

$$= \Delta(a)\ \Delta(\lambda)M^{-1}\ [I - \Delta(q)]^{-1}\ M - \Delta(a)\ \Delta(\lambda)$$

$$= \Delta(a)\ B - \Delta(a)\ \Delta(\lambda)\ , \tag{1.80}$$

where

$$B = \Delta(\lambda)\ M^{-1}\ [I - \Delta(q)]^{-1}\ M\ .$$

Note that

$$\pi M B = \pi T M$$

$$= \pi M$$

and

$$Be' = \Delta(\lambda) M^{-1} [I - \Delta(q)]^{-1}$$

$$= M^{-1} M\Delta(\lambda)M^{-1} [I - \Delta(q)]^{-1} e'$$

$$= M^{-1} T e'$$

$$= M^{-1} e'$$

$$= e' \tag{1.81}$$

These two relations are essentially constraint relations placed on the choice of M and B since they do not depend on the data. It is seen from above that

$$\Delta(\lambda)M^{-1} [I - \Delta(q)]^{-1} e' = e'$$

or, equivalently,

$$[I - \Delta(q)]^{-1} e' = M[\Delta(\lambda)]^{-1} e' ; \tag{1.82}$$

i.e.

$$\frac{1}{1 - q_i} = \sum_{j=1}^{N} M_{ij}/\lambda_j \quad . \tag{1.83}$$

Therefore, the steps involved in identifying the model are.

1) Measure $P(1) = \sum_{i=1}^{N} \bar{\pi}_i q_i$

2) Measure $F(m+1) = \frac{1}{P(1)} \sum_{i=1}^{N} a_i \lambda_i^m$ or, equivalently,

measure $P(0^m 1) = \sum_{i=1}^{N} a_i \lambda_i^m$. From these measurements

determine $\{a_i\}$ and $\{\lambda_i\}$.

3) Measure $F(m+1;\ n+1) = \frac{1}{P(1)} \sum_{i=1}^{N} \sum_{j=1}^{N} a_{ij} \lambda_i^n \lambda_j^m$

for a set of at least N(N-1) different values of m and n and use the constraint relation $Be' = e'$ to obtain a set of linear equations for B (or A).

4) Obtain the model matrix M and the vector q from

$[\Delta(\lambda)]^{-1} B = M^{-1} [I - \Delta(q)]^{-1} M$

and

$Me' = e'$.

5) Obtain the vector π from $P(1) = \sum_{i=1}^{N} \pi_i q_i$.

Note that Step 3 may be replaced by measuring

$$F(m+1 \mid n+1) = \sum_{j=1}^{N} a_j(n) \lambda_j^m \tag{1.84}$$

for at least N-1 values of n to determine the values of

$$a(n) = (a_1(n), \ldots, a_N(n))$$

$$= \frac{e[\Delta(\lambda)]^n A}{e\Delta(a)[I - \Delta(\lambda)] [\Delta(\lambda)]^n e'}$$

for $n = n_1, n_2, \ldots, n_{N-1}$. Solve for A using the constraint $Be' = e'$.

Another useful substitution for Step 3 is to measure

$$F(m+1 \mid n_j \le n \le n_{j+1}) = \sum_{n=n_j}^{n_{j+1}} F(m+1;\ n+1) \left[F(n_j+1) - F(n_{j+1}+1)\right]^{-1}$$

$$= \sum_{i=1}^{N} a_i(n_j, n_{j+1}) \lambda_i^m , \qquad (1.85)$$

where

$$a_i(n_j, j_{n+1}) = \sum_{n=n_j}^{n_{j+1}-1} \frac{\sum_k \sum_\ell a_{k\ell} \lambda_k^n}{\sum_k a_k \left(\lambda_k^{n_j} - \lambda_k^{n_{j+1}}\right)}$$

$$= \sum_{n=n_j}^{n_{j+1}-1} \frac{e[\Delta(\lambda)]^n A}{e\Delta(a)\left\{[\Delta(\lambda)]^{n_j} - [\Delta(\lambda)]^{n_{j+1}}\right\}} \qquad (1.86)$$

The procedure for using this approach is the same as in Step 3.

(c) Properties of Error Detecting Codes

A number of texts such as Peterson [17] and Liu [18] discuss basic properties of error detecting codes. However, the main thrust of these texts, and, in fact, of recent work in coding theory, seems to be the study of error correcting, rather than error detecting, codes. Although tractable relations between the error detecting and error correcting properties of codes are well known, a good error correcting code is not necessarily a good error detecting code.

Only a few papers devoted to error detecting codes, such as those by Corr [19] and by Peterson [20], were found in the review of the literature. Whereas synthesis procedures were found for error correcting codes, none could be found for error detecting codes.

This section of the report summarizes material from the literature, (principally Peterson [17] and Liu [18]), pertaining to basic properties of error detecting codes which are germane to the remainder of the study. Attention is restricted to linear, binary, cyclic, block codes.

In the present context an encoder maps a sequence of binary message digits into a sequence of binary code digits. The message and its code word image both have fixed lengths for the type of codes being considered and hence they can be regarded as vectors. Consider a message vector of k digits. A code vector of n digits is formed to correspond to each message vector. The code vector can be constructed in a "systematic form" consisting of the k message digits preceeded (or followed) by n - k redundant digits. The problem of code design amounts to finding an algorithm for choosing the n - k redundant digits in the code vector so that error detection, or error correction, is carried out with the smallest possible probability of error.

In the study of linear binary cyclic codes it is convenient to treat the components of code and message vectors as coefficients of a polynomial. This results in a one-to-one correspondence between, for example, a code vector v and a code polynomial V(X) as given by

$$v = (v_0, v_1, \ldots, v_{n-1}) \Leftrightarrow V(X) = v_0 + v_1 X + \ldots + v_{n-1} X^{n-1} \tag{1.87}$$

A similar correspondence is set up for message vectors. Using this artifice, it is possible to investigate the structure of codes through a study of appropriate binary polynomials.

Some of the more important properties of codes, with respect to the present study, will be summarized below in terms of these binary polynomials. Proofs of the properties will be found in the references, particularly [18] and [20].

Every code polynomial V(X) in a (n, k) cyclic code can be expressed as

$$V(X) = M(X)\ g(X) \tag{1.88}$$

where

$$M(X) = m_0 + m_1 X + \ldots + m_{k-1} X^{k-1} \tag{1.89}$$

can be the message polynomial and

$$g(X) = 1 + g_1 X + g_2 X^2 + \ldots + g_{n-k-1} X^{m-k-1} + X^{n-k} \tag{1.90}$$

is termed a "code generator" polynomial.

In a (n, k) cyclic code there exists one and only one generator polynomial, g(X), of degree n - k. (The degree of a polynomial is the largest power of X in a term with a nonzero coefficient.) Every code polynomial, $V(X)$, is a multiple of $g(X)$ and every polynomial of degree $n - 1$ or less which is a multiple of $g(X)$ must be a code polynomial. Therefore the code is completely specified by the generator polynomial, $g(X)$.

If $a(X)$, $b(X)$ and $c(X)$ are polynomials and

$$a(X)\ b(X) = c(X),$$

then $a(X)$ and $b(X)$ are said to be "factors" of $c(X)$ or $c(X)$ is divisible by $a(X)$ and $b(X)$. A polynomial $p(X)$ of degree n greater than 0 which is not divisible by any polynomial of degree less than n is called "irreducible."

The generator polynomial of a (n, k) cyclic code is a factor of $X^n + 1$, i.e.

$$X^n + 1 = g(X)\ h(X). \tag{1.91}$$

Conversely, if $g(X)$ is a polynomial of degree $n - k$ and is a factor of $X^n + 1$, then it generates a (n, k) cyclic code.

An irreducible binary polynomial of degree m is "primitive" if and only if it devides $X^n + 1$ for n no less than $2^m - 1$. Thus a primitive polynomial of degree $n - k$ will divide $X^n + 1$ for n no less than $2^{n-k} - 1$ and hence generates a code of length at least $2^{n-k}-1$. A code generated by a primitive polynomial is called a Hamming code.

The class, $\{V\}$, of code vectors for a binary, cyclic (n, k) code generated by $g(X)$ has the properties:

i. $\{V\}$ contains the zero vector
ii. $\{V\}$ contains the sum of any two vectors in $\{V\}$
iii. if $V_1 = (v_0, \ldots, v_n)$ is in $\{V\}$ then so is $V_2 = (v_{n-\ell}, \ldots, v_0, \ldots, v_{n-\ell-1})$ for $\ell = 1, 2, \ldots, n$.

Code vectors can be expressed in the systematic form

$$V(X) = R(X) + X^{n-k} M(X). \tag{1.92}$$

Since, $V(X) = g(X)\, Q(X)$, (1.92) can be written as

$$g(X)\, Q(X) = R(X) + X^{n-k} M(X) \tag{1.93}$$

showing that $R(X)$ can be constructed as the remainder resulting from the division of $X^{n-k} M(X)$ by $g(X)$. Note that the vector corresponding to the polynomial of (1.92) is

$$V(X) \Leftrightarrow v = (v_0, v_1, \dots v_{n-k-1}, m_0, m_1, \dots m_{k-1}) \tag{1.94}$$

the systematic form of the code vector with $n - k$ check bits, v_i, followed by k message bits, m_i.

A "shortened" code results if all the code vectors having z higher order information digits equal to zero (i.e. $m_{k-1} = m_{k-2} = m_{k-z-n} = 0$) are deleted from $\{V\}$. The result is a linear $(n - z, k - z)$ code which is not cyclic. Note that the code vector set of the shortened code is $\{V\}$ with some code vectors deleted.

Let the received code vector after transmission through some channel be denoted $W(X)$. Then $W(X)$ is given by

$$W(X) = V(X) + E(X) \tag{1.95}$$

where $E(X)$ is a polynomial corresponding to the vector of additive errors introduced by the channel.

Error detection is achieved by observing the "syndrome", $S(X)$, which is the remainder resulting from dividing $W(X)$ by $g(X)$. Since $W(X)$ is the sum of $V(X)$, (which is a multiple of $g(X)$), and $E(X)$, $S(X)$ will be zero for the case of no errors for which $E(X) = 0$. Unfortunately

S(X) is also zero if E(X) is some multiple of g(X), in which case there are "undetectable errors." Note that the class $\{V(X)\}$ of code vectors is generated by multiples of g(X). Therefore the class of undetectable error vectors is identical with the class of code vectors $\{V(X)\}$.

The following are some error detecting properties of cyclic codes:

All single errors are detected if g(X) has more than one term.

If g(X) contains a factor $1 + X^c$, any odd number of errors will be detected.

A code generated by g(X) detects all single and double errors if the length n of the code is no greater than the exponent e to which g(X) belongs. (g(X) belongs to exponent e if e is the least positive integer such that g(X) evenly divides $X^e + 1$.).

For any m there is a double error detecting (Hamming) code of length $n = 2^m - 1$ generated by a g(X) of degree m.

Any cyclic code generated by a g(X) of degree n - k detects any error burst of length n - k or less.

The fraction of bursts of length $b > n - k$ that are detected is

$$2^{-(n-k-1)} \text{ if } b = n - k+1$$

$$2^{-(n-k)} \text{ if } b > n - k+1$$

Cyclic (Fire) codes generated by

$$g(X) = (X^c + 1)\, g_1(X)$$

will detect any combination of two bursts if:

(i) $c + 1 \geq$ sum of burst lengths

(ii) $g_1(X)$ is irreducible and a degree at least as great as the length of the shorter burst

(iii) $n \leq$ least common multiple of c and the exponent e to which $g_1(X)$ belongs

An important class of codes, which will be used in Part II of this report, are referred to as BCH codes. These codes can be constructed in a systematic manner. For any choice of m and t there exists a BCH code of length $2^m - 1$ which is guaranteed to detect any combination of $2t$ errors. The generator polynomial of such a code is of degree no greater than mt.

The procedure for constructing the most important type of BCH codes, referred to as narrow-sense or primitive BCH codes, is the following.

Let α be a root of a primitive polynomial of degree m. The polynomial m(X), which is the binary polynomial of smallest degree for which $m(\alpha) = 0$, is referred to as the "minimal polynomial" of α. Consider the sequence $\alpha, \alpha^2, \alpha^3, \ldots \alpha^{2t}$ of consecutive powers of α and denote by $m_i(X)$ the minimum polynomial of α^i. Then the generating polynomial of a 2t-error-detecting BCH code is the least common multiple of $m_1(X)$, $m_2(X), \ldots m_{2t}(X)$. Since it can be shown that every even power of α has the same minimum polynomial as some previous odd power, the generating polynomial can be expressed concisely as

$$g(X) = \text{LCM}\,[m_1(X), m_3(X), \ldots m_{2t-1}(X)] \tag{1.96}$$

The degree of each minimal polynomial $m_i(X)$ constructed as indicated from α, which is a root of a primitive polynomial of degree m, is m or less. Thus the degree of g(X) is at most mt.

Tables of primitive and minimal polynomials of various degrees are available in the literature. Perhaps the most widely used table is found in Peterson [17], pp. 472 - 492. This table lists all irreducible polynomials (including primitive polynomials) of degree 16 or less and a primitive polynomial with a minimum number of nonzero coefficients and polynomials belonging to all possible exponents for each degree 17 through 34. For each degree, m, the table lists a primitive polynomial with a minimum number of nonzero coefficients. Denoting α as a root of this primitive polynomial, the table also lists minimum polynomials of α^j for j odd.

To illustrate the use of the table in constructing BCH codes, consider the problem of constructing a code of length at least 2000 bits with 32 check bits which can detect as many errors as possible.

The degree of g(X) is equal to the number of check bits and the length of the code is $2^m - 1$. This results in the constraints

$$32 = mt$$

$$2000 \geq 2^m - 1.$$

Since 2^{11} - 2048 and $2^{10} = 1024$, the last constraint forces m to be greater than or equal to 11. If there are to be exactly 32 check bits, then $t = 1$, $m = 32$ and $t = 2$, $m = 16$ are possible combinations.[†] For $t = 1$ any primitive 32 degree polynomial from the table would serve for g(X). For $t = 2$,

$$g(X) = \text{LCM}\,[m_1(X),\ m_3(X)]$$

where $m_1(X)$ is a primitive 16th degree polynomial selected from the table. If α is a root of $m_1(X)$, then $m_3(X)$ is the minimum polynomial of α^3, a polynomial which can also be found in the table.

[†] These codes are guaranteed to detect any combination of 2t errors since a shortened cyclic code has at least as great a minimum distance as the cyclic code from which it is derived and it can detect any burst-error patterns that the original code could detect.

Since the second code with $t = 2$ has the greater guaranteed error protection, one would be inclined to chose it. However, a principle result of this research is to show that this approach is not the best for real channels.

3. Channel Models Chosen for the Code Study

Renewal models were chosen for the code study for three reasons, namely: they accurately approximate HF and wireline channels which are important in the Defense Communications system, data has been compiled and used to determine the parameters of such models to match practical systems and finally work with nonrenewal models in terms of both theory and the necessary practical data does not seem to be sufficiently advanced to justify a general code study based on these models.

Ten renewal models were chosen for the study, namely:

a) The Pareto model used by Johnson [21] and developed by Bolkovic et.al. [28] for a switched telephone network.

b) A model termed the Markov-Fritchman model developed for an HF link.

c) A model termed the Markov-Tasi model developed for a different HF link from that of (b).

d) Seven models developed by Brayer [26] to match experimental data from the AUTOVON system.

The models developed by Brayer are part of an extensive study done by MITRE in conjunction with the DICEF facility at RADC. Brayer's report [26] should be consulted for the details of developing the models. Generally speaking the experimental data was taken from parts of the continental AUTOVON system involving two to five switches at data rates of 4800 b/s and 9600 b/s. A total of approximately 20,000 error bursts of length greater than 32 bits was found in the data with approximately 5000 of these bursts in the 4800 bit/sec data and approximately 15,000 in the 9600 bit/sec data.

A summary of the models is given below:

Pareto Model $$F(n+1) = \frac{(1+n)^{-\alpha} - L^{-\alpha}}{(1 - L^{-\alpha})}, \quad 0 \le n \le L-1$$

$$L = \left[\frac{1-\alpha}{\alpha}(E(n)+1)\right]^{\frac{1}{1-\alpha}}$$

$$\alpha = 0.3, \; E(n) = 3 \times 10^4$$

Markov-Fritchman

$$T = \begin{bmatrix} 0.66 & 0 & 0.34 \\ 0 & 0.9991 & 0.0009 \\ 0.44 & 0.34 & 0.22 \end{bmatrix}$$

Markov-Tsai

$$T = \begin{bmatrix} 0.99911 & 0 & 0.00089 \\ 0 & 0.73644 & 0.26356 \\ 0.36258 & 0.58510 & 0.05232 \end{bmatrix}$$

Brayer Table 3 (two switches - 4800 b/s)

$$T = \begin{bmatrix} 0.9754047 & 0.0 & 0.0 & 0.0245953 \\ 0.0 & 0.9995566 & 0.0 & 0.0004434 \\ 0.0 & 0.0 & 0.9999969 & 0.0000031 \\ 0.5131625 & 0.2505878 & 0.0895789 & 0.1466708 \end{bmatrix}$$

$P(1) = 3.39 \times 10^{-5}$

Brayer Table 4 (three switches - 4800 b/s)

$$T = \begin{bmatrix} 0.2156599 & 0.0 & 0.0 & 0.0 & 0.0 & 0.7843401 \\ 0.0 & 0.8886233 & 0.0 & 0.0 & 0.0 & 0.1113767 \\ 0.0 & 0.0 & 0.9987018 & 0.0 & 0.0 & 0.0012982 \\ 0.0 & 0.0 & 0.0 & 0.9999393 & 0.0 & 0.0000607 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.9999977 & 0.0000023 \\ 0.1124190 & 0.1780878 & 0.1994085 & 0.2057504 & 0.0209361 & 0.2833981 \end{bmatrix}$$

$P(1) = 7.93 \times 10^{-5}$

<u>Brayer Table 5</u> (four switches - 4800 b/s)

$$T = \begin{bmatrix} 0.9611693 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0388307 \\ 0.0 & 0.8898716 & 0.0 & 0.0 & 0.0 & 0.1101284 \\ 0.0 & 0.0 & 0.9988276 & 0.0 & 0.0 & 0.0011724 \\ 0.0 & 0.0 & 0.0 & 0.9999507 & 0.0 & 0.0000483 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.9999969 & 0.0000031 \\ 0.2044374 & 0.2271502 & 0.0585721 & 0.0422230 & 0.0166381 & 0.4509793 \end{bmatrix}$$

$P(1) = 1.58 \times 10^{-4}$

<u>Brayer Table 6</u> (five switches - 4800 b/s)

$$T = \begin{bmatrix} 0.9068574 & 0.0 & 0.0 & 0.0 & 0.0931426 \\ 0.0 & 0.9980199 & 0.0 & 0.0 & 0.0019801 \\ 0.0 & 0.0 & 0.9999507 & 0.0 & 0.0000493 \\ 0.0 & 0.0 & 0.0 & 0.9999981 & 0.0000019 \\ 0.3606889 & 0.0668155 & 0.0379466 & 0.0305986 & 0.5039504 \end{bmatrix}$$

$P(1) = 6.05 \times 10^{-5}$

<u>Brayer Table 7</u> (two switches-9600 b/s)

$$T = \begin{bmatrix} 0.9995636 & 0.0 & 0.0004364 \\ 0.0 & 0.9999922 & 0.0000078 \\ 0.4874004 & 0.1026200 & 0.4099796 \end{bmatrix}$$

$P(1) = 7.01 \times 10^{-5}$

<u>Brayer Table 8</u> (three switches-9600 b/s)

$$T = \begin{bmatrix} 0.9982991 & 0.0 & 0.0017119 \\ 0.0 & 0.9999714 & 0.0000286 \\ 0.3635153 & 0.2268990 & 0.4095857 \end{bmatrix}$$

$P(1) = 1.23 \times 10^{-4}$

<u>Brayer Table 9</u> (four switches-9600 b/s)

$$T = \begin{bmatrix} 0.9999391 & 0.0000609 \\ 0.3979892 & 0.6020108 \end{bmatrix}$$

$P(1) = 1.52 \times 10^{-4}$

Useful relations for renewal models of several types, including those chosen for further study, are summarized in Table 1.1.

$$P(m,n) = \sum_{j=1}^{n-m+1} P(1)F(j)R(m,n-j+1) , \qquad 1 \le m \le n$$

$$R(m,n) = \begin{cases} F(n) & 1 = m \le n \\ \sum_{j=1}^{n-m+1} p(j)R(m-1,n-j) & 2 \le m \le n \end{cases}$$

$$a(j) = \begin{cases} 1 & j = 0 \\ p(j) + \sum_{i=1}^{j-1} p(i)a(j-i) & j \ge 1 \end{cases}$$

$$\Leftrightarrow P(m,n) = \begin{cases} \sum_{k=1}^{n} P(1)F(k)F(n-k+1), & 1 = m \le n \\ \sum_{k=1}^{n-m+1} p(k)P(m-1,n-k), & 2 \le m \le n \end{cases}$$

Channel Model	$I(1)$	$F(m+1)$
Binary Symmetric	q	$(1-q)^m$
Gilbert	$\dfrac{q\, t_{12}}{t_{12} + t_{21}}$	$\left[t_{11}+t_{22}(1-q)\right] F(m)-(1-q)(t_{11}-t_{21})F(m-1), 2 \le m$
Pareto	$\dfrac{1-\alpha}{\alpha} L^{\alpha-1}$	$1 - \dfrac{1}{1-L^{-\alpha}} \left[\dfrac{(m+1)^{\alpha} - 1}{(m+1)^{\alpha}}\right] , \quad 0 \le m \le L - 1$
Markov Models	$\left[1 + \sum_{k=1}^{N-1} \dfrac{t_{Nk}}{1 - t_{kk}}\right]^{-1}$	$\sum_{k=1}^{N-1} t_{Nk} \left(t_{kk}\right)^{m-1} , \quad m \ge 1$

TABLE 1.1 Summary of Renewal Models

4. References

1. E. O. Elliott, "Estimates of Error Rates for Codes on Burst Noise Channels," Bell Syst. Tech. J., Vol. 42, Sept. 1963, pp. 1977-1997.

2. E. O. Elliott, "A Model of the Switched Telephone Network for Data Communications," Bell Syst. Tech. J., Vol. 44, Jan. 1965, pp. 85-109.

3. B. D. Fritchman, "A Binary Channel Characterization using Partitioned Markov Chains," IEEE Trans. Inform. Theory, Vol. IT-13, April 1967, pp. 221-236.

4. E. N. Gilbert, "Capacity of a Burst Noise Channel," Bell Syst. Tech. J., Vol. 39, September 1960, pp. 1253-1265.

5. J. P. Adoul, B. D. Fritchman, and L. N. Kanal, "A Critical Statistic for Channels with Memory," IEEE Trans. Inform. Theory, Vol. IT-18, Jan. 1972, pp. 133-141.

6. S. Tsai, "Simple Partitioned Markov Chain Model and Troposcatter Channel," NTC-73, Atlanta, Georgia, November 1974, pp. 16F-1 - 16F-6.

7. S. Tsai, "Markov Characterization of the HF Channel," IEEE Trans. on Communications Technology, Vol. COM-17, Feb. 1969, pp. 24-32.

8. S. Tsai, "Analytic Method for Evaluating Error Correcting Codes on a Real Channel," ICC-72.

9. R. T. Chien, A. H. Haddad, B. Goldberg, and E. Meyers, "An Analytic Error Model for Real Channels," ICC-72, Philadelphia, Pa., June 1972.

10. J. M. Berger and B. Mandelbrot, "A New Model for Error Clustering in Telephone Circuits," IBM J. Res. Div., Vol. 7, July 1963, pp. 224-236.

11. S. M. Sussman, "Analysis of the Pareto Model for Error Statistics on Telephone Circuits," IEEE Trans. on Commun. Systems, Vol. CS-11, June 1963, pp. 213-221.

12. R. G. Gallager, Information Theory and Reliable Communication, Wiley, New York, 1968 (Chapter 6).

13. M. Munter and J. K. Wolf, "Predicted Performances of Error-control Techniques over Real Channels," IEEE Trans. Inform. Inform. Theory, Vol. IT-14, Sept. 1968, pp. 640-650.

14. H. A. Blank and P. J. Trafton, "A Markov Error Channel Model," NTC-73, Atlanta, Georgia, November 26-28, 1973, pp. 15B-1 - 15B-8.

15. P. A. Bello, "A Troposcatter Channel Model," IEEE Trans. on Communication Technology, Vol. COM-17, April 1969.

16. J. Goldman, "Multiple Error Performance of PSK Systems with Cochannel Interference and Noise," IEEE Trans. on Communication Technology, Vol. COM-19, August 1971, pp. 420-430.

17. W. W. Peterson and E. J. Weldon, Jr., Error Correcting Codes, (Second Edition), MIT Press, Cambridge, 1972.

18. S. Liu, An Introduction to Error-Correcting Codes, Prentice-Hall, Englewood Cliffs, 1970.

19. F. P. Corr, "Statistical Evaluation of Error Detection Cyclic Codes for Data Transmission," IEEE Trans. on Communications Systems, June 1964, pp. 211-216.

20. W. Peterson and D. Brown, "Cyclic Codes for Error Detection," Proc. IRE Vol. 49, Jan. 1961, pp. 228-235.

21. D. C. Johnson, "Performance of the SDLC, CRC-16 and CCITT Polynomials with NRZI Coding," X3S34 and CRC Ad Hoc Group Memo, April 1972.

22. R. E. Townsend and T. N. Watts, "Effectiveness of Error Control in Data Communication over the Switched Telephone Network," BSTJ, Vol. 43, Nov. 1964.

23. K. Brayer and O. Cardinale, "Evaluation of Error Correcting Block Encoding on High-Speed HF Data," IEEE Trans. Comm. Tech, June 1967, pp. 371-382.

24. K. Brayer, "Error Patterns Measured on Transequatorial HF Communication Links," IEEE Trans. Comm. Tech, April 1968.

25. K. Brayer, "Error Correction Code Performance on HF, Troposcatter, and Sattellite Channels," IEEE Trans. Comm. Tech, Vol. Com-19, Oct. 1971, pp. 781-789.

26. K. Brayer, "Characterization and Modeling of the Digital High-Speed Autovon Channel," Interim Report MTR2802, 1 May 1974 MITRE Corp.

27. R. T. Chien, et.al., "Analytical Mathematical Models of Tactical Military Communications Channels," Quarterly Report, September 1973, Research and Development Technical Report ECOM-0292-7.

28. M. D. Balkovic, et.al., "High-Speed Data Transmission Performance on the Switched Telecommunications Network," B.S.T.J., April 1971.

29. J. Salz and B. R. Saltzberg, "Double Error Rates in Differentially Coherent Phase Systems," IEEE Trans. on Communication Systems, Vol. CS-12, June, 1964, pp. 202-205.

30. R. Ash, Information Theory, New York: Interscience Publishers, 1965.

31. B. D. Fritchman, "A Binary Channel Characterization Using Partitioned Markov Chains with Applications to Error Correcting Codes," Ph.D. Thesis, Lehigh University, 1967.

PART II

CODE EVALUATION USING RENEWAL CHANNEL MODELS

1. Probability of Undetectable Errors for Renewal Channel Models

As discussed in Part I, several channel models are discussed by a number of authors, typical references are [1] and [10]. The basic assumption of renewal models is that the "gap" intervals between errors are independent random variables. Figure 2.1 illustrates the definition of gap length, d, as one plus the number of nonerrors between two errors.

$$1 \quad \underbrace{\overbrace{0 \;\; 0 \;\; \cdots \;\; 0 \;\; 1}^{\text{error gap}}}_{\longleftarrow\ d\ \longrightarrow} \quad 0 \;\; \cdots$$

Figure 2.1 A Typical Error Gap

Two gap statistics are useful, namely:

$p(d) = P\{0^{d-1}\,1|1\}$ - the probability of exactly d-1 nonerrors followed by an error in the error pattern, given an error starting the pattern.

$F(d) = P\{0^{d-1}\;|1\}$ - the probability of at least d-1 nonerrors followed by an error, given an error starting the pattern.

The two statistics are related by the equations

$$F(d) = \sum_{k=d-1}^{\infty} P\{0^k\,1|1\} = \sum_{k=d}^{\infty} p(k) \tag{2.1}$$

$$p(d) = F(d) - F(d+1)\ . \tag{2.2}$$

A central objective in the study of error detecting codes is an evaluation of the probability, $P_u(n)$, of undetected error for a particular code for blocks of length n. Techniques are available for identifying undetectable error patterns for given codes. Given a particular error pattern, e, its probability, P(e), can be computed for a particular channel model. The sum of the probabilities of all undetectable error patterns is the undetected error probability for the code based on the assumed channel model.

Figure 2.2 shows a particular undetectable error pattern, e, for a block of length n.

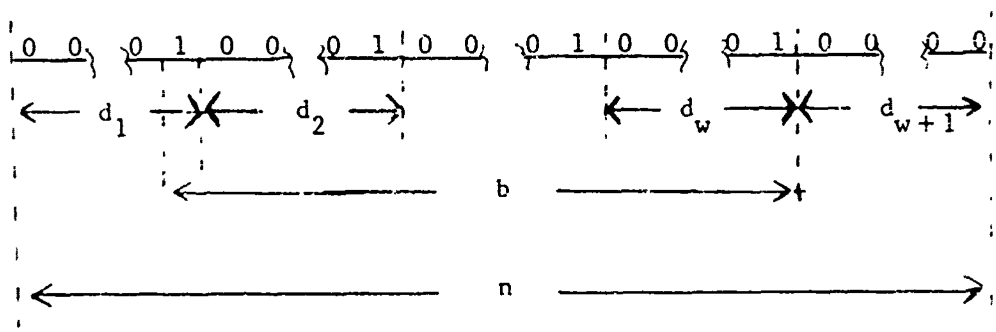


Figure 2.2 Undetectable Error Pattern

It is useful to identify the "burst length," b, containing all of the errors and a particular "burst pattern," e_b, beginning with the first error and ending with the last error.

For a cyclic code an undetectable error pattern will result from every position of the burst pattern within the block n. Thus undetectable error patterns exist for every d_1 in the interval $1 \leq d_1 \leq n - b + 1$, where d_{w+1} is constrained to satisfy

$$d_{w+1} = n - b + 1 - d_1 \ . \tag{2.3}$$

The probability of the pattern, e, of Figure 2.2 can be computed as follows. The internal gaps of length d_2, d_3, $\cdots$ d_w each have a probability given by $P(d_i)$. Since the gap lengths are independent for renewal models, the probability, $P_g[e_b(w,b)]$, of the internal gap pattern in the burst is given by

$$P_g[e_b(w,b)] = \prod_{i=2}^{w} P(d_i) , \tag{2.4}$$

where each d_i has a value corresponding to the particular pattern, e_b. Note that w is the number of errors, or the weight, of the burst pattern.

The probability of the gap beginning the pattern is the probability of at least $d_1 - 1$ zeros in the error pattern followed by a one. Note that the one starting the gap is at an unspecified position outside the block being considered. The probability of the beginning gap can be expressed as

$$P\{\cdots\cdot\ 00\ \cdots\ 01\} = P\{0^{d-1}\ |1\}\ P(1) = P(1)\ F(d_1) \tag{2.5}$$

using the relation for conditional probability. Similarly the probability of the ending pattern is just the probability of at least d_{w+1} zeros given a one to start the pattern, or $F(d_{w+1} + 1)$. Note that in this case the one ending the pattern is outside of the block being considered.

Since the gaps beginning and ending the block are statistically independent of the others for a renewal channel, the probability, $P(e)$, of all of the gaps in a particular pattern is given by

$$P(e) = P(1)\ F(d_1)\ F(d_{w+1} + 1)\ P_g[e_b(w,b)] \tag{2.6}$$

The total probability, $P(e_b;n)$, of all undetectable error patterns which include the burst pattern e_b in all of its possible positions* can be expressed as

$$P(e_b;n) = P(1)\ P_g[e_b(w,b)] \sum_{d_1=1}^{n-b+1} F(d_1)\ F(n - b + 2 - d_1) \tag{2.7}$$

Note that the total probability is just the sum of the separate pattern probabilities since the patterns are mutually exclusive.

---

* Code vectors severely truncated from their "natural" length are typical in the present study. Thus patterns shifted to fold over the end of the block are not considered.

It is useful to define a variable $S_p(n,b)$ by the equation*

$$S_p(n,b) = \frac{1}{n} \sum_{d_1=1}^{n-b+1} F(d_1)\, F(n - b - d_1 + 2), \tag{2.8}$$

so that $P(e_b;n)$ is expressed simply as

$$P(e_b;n) = n\, P(1)\, P_g[e_b(w,b)]\, S_p(n,b). \tag{2.9}$$

The probability of undetectable errors for a particular code is obtained by summing over the probability of all undetectable patterns for the code. Equation (2.9) gives the probability of all cyclical shifts of a pattern with: (i) fixed burst length, b, (ii) fixed weight, w, and (iii) a fixed distribution of the errors within the burst, as specified by fixed d_i, i=2, ... w. To obtain the total probability of undetectable errors for a given code, probabilities $P(e_b;n)$ must be computed and summed over the class, W, of all of the variables listed above, namely: all burst lengths, all weights and all distributions of errors of fixed weight within a given burst length. The result can be expressed as

$$P_u(n) = n\, P(1) \sum_W S_p(n,b)\, P_g[e_b(w,b)] = \sum_W P(e_b;n) \tag{2.10}$$

2. <u>Approaches to Code Evaluation</u>

Probability of undetected error is the chief measure of the quality of an error detecting code. In principle for a given code and channel model all undetectable error patterns can be identified, the probability of each can be computed and the probability of undetected error obtained from (2.10). The difficulty with this procedure is the fact that if the number of message bits is k, then there are 2^k undetectable error patterns. The last statement follows from the fact that the set of undetectable error patterns is identical with the set of code vector patterns.

---

* This definition is suggested by Johnson [21] in an unpublished memo.

For example if the block length is 2000 and there are 32 check bits, k is 1968 and 2^k is 10^{592}, a number too large to permit computation of all pattern probabilities.

Johnson in his unpublished memo [21] estimates the probability of undetected error by computing and summing the probabilities of undetectable error patterns with fairly short bursts and relatively few errors. Johnson's computational algorithm requires a search through all patterns of fixed length and weight to find the undetectable patterns. Computing time limits such a search to weights on the order of 6 and less and bursts of length on the order of 100 bits. Using 10 to 15 minutes of large general purpose computer time, thirty to fifty undetectable error patterns can be found and processed in this way to produce an estimate of the probability of undetectable error.

3. Development of an Efficient Algorithm for Code Evaluation

Consider (2.10) which expresses the probability of undetectable errors for a given code, a given block length and a given channel model. In particular consider the quantity, $S_p(n,b)$, in this equation. Curves of $S_p(n,b)$ versus b have been computed for a number of renewal channel models with different choices for the gap distribution function and the results are given in Appendix I as Figures A.17-A.20. Examination of these curves shows empirically that, (at least for the models considered), $S_p(n,b)$, can be approximated by a constant, $\overline{S}_p(n)$, which is independent of b.

Thus a reasonable approximation for $P_u(n)$ is given by

$$P_u(n) \cong n\ P(1)\ \overline{S}_p(n) \sum_{W} P_g\ [e_b(w,b)] \tag{2.11}$$

It also follows from (2.10) that exact upper and lower bounds on $P_u(n)$ are given by

$$n\,P(1)\,S_p^{*}(n) \sum_W P_g \le P_u(n) \le n\,P(1)\,S_p^{**}(n) \sum_W P_g \tag{2.12}$$

where

$$S_p^{*}(n) = \min_b S_p(n,b) \tag{2.13}$$

$$S_p^{**}(n) = \max_b S_p(n,b) \tag{2.14}$$

Note in (2.11) that $P_u(n)$ is expressed as the product of the term $n\,P(1)\,\overline{S}_p(n)$, which is independent of the code, and the term $\sum_W P_g$ which depends on the code. The bounds in (2.12) break up into two terms in a similar way with $\sum_W P_g$ again being the code dependent term.

In comparing two codes with respect to probability of undetectable errors, it thus seems reasonable to use $\sum_W P_g$ as a figure of merit. The figure of merit is proportional to probability of undetectable errors, or bounds on this quantity, for a fixed block length, n, and a fixed channel model.

A tractable algorithm for computing an approximation to $\sum_W P_g$ is now developed. First consider the expression

$$\sum_W P_g = \sum_{W_1} P_g + \sum_{W_2} P_g \tag{2.15}$$

which partitions the sum over all undetectable error patterns into two parts. The quantity $\sum_{W_1} P_g$, summing the probability of selected undetectable error patterns, will be used to approximate $\sum_W P_g$. The set W_1 will be chosen to include all of the high probability error patterns so that $\sum_{W_2} P_g$ is made negligible in comparison to $\sum_{W_1} P_g$.

To specify the set W_1, consider the code vector pattern, or equivalently the undetectable error pattern, with γ check bits shown in Figure 2.3

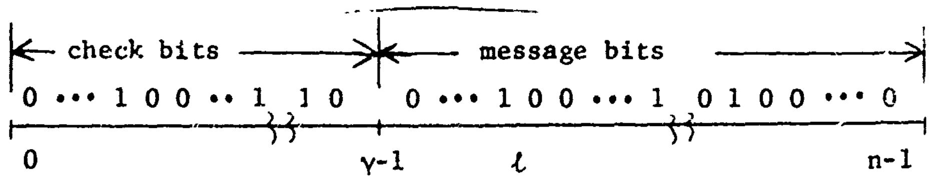


Figure 2.3 A Typical Code Vector Pattern with γ Check Bits

For the pattern of Figure 2.3, P_g is determined as the product of the probabilities of the specified gaps beginning after the first error and continuing to include the last error, as expressed in (2.4).

The error gaps involved in computing P_g can be classified as contributing to three probabilities, namely:

- P_C - probability of the gaps in the check bit portion of the pattern
- P_T - probability of the transition gap between the check bits and the message bits
- P_M - probability of gaps in the message portion of the pattern.

Thus P_g is expressed as

$$P_g = \prod_{i=2}^{w} P(d_i) = P_C \, P_T \, P_M \, . \tag{2.15}$$

Since $\sum_W P_g$ contains a term for every possible message pattern, the message patterns can play the role of an independent variable in constructing undetectable error vectors through use of standard coding algorithms. Furthermore, from (2.15), it can be noted that large values of P_g will result if both P_M and $P_C \, P_T$ are large. Large P_M is a necessary but not sufficient condition for a large P_g.

Now consider a set, $\hat{W}_1$, of message patterns constructed so that for each pattern

$$P_M \geq \beta^* \, . \tag{2.16}$$

The patterns in this set can be used to construct code vectors which satisfy the necessary condition for large P_g.

With reference to the typical code vector pattern of Figure 2.3, note that only the probabilities of the gaps within the error burst, (i.e. the bits following the first error and extending to and including the last error), effect P_g by the way in which it is defined. Furthermore, since the code is cyclic, every possible shifted position of a basic pattern will also appear in a code vector. The cyclic shifts of a fixed basic pattern are accounted for by the factor $\overline{S}_p(n)$ in (2.11) and hence only one position of a basic pattern should be included in a final set W_1. This is accomplished by including in W_1 only those code vectors of $\hat{W}_1$ which begin with a one in the first position.

Relative to constructing the message pattern set, consider the case for which the first one in a message pattern being considered is located at ℓ as shown in Figure 2.3. The combined length of the gaps in the check bit pattern and the transition gap is thus ℓ.

It is convenient in computational work to construct the class of message vectors so that

$$p(\ell)\ P_M \geq \beta \ , \tag{2.18}$$

which partially accounts for the transition gap in setting the bound. (It can be shown that $p(\ell)$ is the maximum probability of the check and transitions gaps, given that the first message bit is located at ℓ.) Since ℓ and hence $P(\ell)$ is determined by each particular message pattern, in theoretical work it is more convenient to maximize $p(\ell)$ over all values of ℓ to obtain $p(\gamma)$ which is independent of the particular message pattern. In such a case message vectors would be constructed to satisfy

$$p(\gamma)\ P_M \geq \beta \ , \tag{2.19}$$

which yields a slightly different class. Either (2.18) or (2.19) can be solved for a bound on P_M and the results can be expressed as

$$P_M \geq \frac{\beta}{p(\ell)} \geq \frac{\beta}{p(\gamma)} = \beta^* \qquad (2.20)$$

It is now possible to state the following efficient algorithm, termed the $\sum P_g$ algorithm, for code evaluation.

$\sum P_g$ Algorithm

Step 1. Find a set, $\hat{W}_1$, of message patterns such that for each pattern $p(\ell)\ P_M \geq \beta$, or alternately $P_M \geq \beta^*$. (This set can be used for any code polynomial but the set depends, weakly, on the distribution function for the error gaps as specified by the channel model.)

Step 2. For a given code polynomial compute the check bit pattern corresponding to each message pattern of step (1).

Step 3. Construct the code vector patterns correspondong to each message in the set $\hat{W}_1$. These patterns are also undetectable error patterns.

Step 4. Discard those patterns which do not begin with a one to obtain a reduced set of patterns, W_1.

Step 5. Compute P_g for each undetectable error pattern of step 4.

Step 6. Compute $\sum_{W_1} P_g$ where the sum extends over all message patterns in the set, W_1, determined in step (4).

Note that through use of (2.11) $\sum_{W_1} P_g$ can be used to compute the following estimate, $\tilde{P}_u(n)$ of $P_u(n)$,

$$\tilde{P}_u(n) = n\ P(1)\ \bar{S}_p(n) \sum_{W_1} P_g. \qquad (2.21)$$

4. Evaluation of $\sum P_g$ Algorithm.

Three sets of message vectors have been constructed according to the data in Table 2.1. Message patterns were generated using the condition of (2.18) for the values of β specified for 16 check bits. The set of message

Table 2.1 Data on Message Patterns

No. of Patterns	β^*	β(16 check bits)	β (32 check bits)
864	7.3×10^{-3}	5.8×10^{-5}	2.4×10^{-5}
6,117	1.3×10^{-3}	1.0×10^{-5}	4.2×10^{-6}
32,362	2.7×10^{-4}	2.1×10^{-6}	8.7×10^{-7}

patterns generated is slightly different than would have been obtained from using the β^* bound and the tabulated values. In the table note that values of β and β^* are related by (2.20).

The number of code vector patterns comprising W_1 is approximately one-half of the number of message patterns in the table since only code vector patterns with ones in the first bit are retained. The smallest set (864 message patterns) includes all patterns with 1, 2 and 3 errors (as well as other patterns) while the largest set includes all patterns with 1, 2, 3, 4 and 5 errors and other patterns.

To satisfy a given bound on P_M, in principle a new set of message sequences should be chosen for each channel model since gap probabilities are specified by the model. However, for any distribution function which assigns uniformly less probability to any given gap than the Pareto distribution, the Pareto message set will also satisfy the given bound.

The gap distribution functions plotted in Appendix 1, Figure A.1, shows that several channel models of interest, such as the Markov-Fritchman model, are bounded by the Pareto distribution so that the Pareto message sets exactly satisify the given bound. In several other cases, while the Pareto distribution does not exactly bound the distribution function of other models, it is approximately equal to several of them over regions where it does not bound. The only case of a substantial difference between the Pareto distribution, either as a bound or as approximately equality, is the case of Brayer Table 3. Even in this case the difference is not an order of magnitude.

In the body of the study only the Pareto message sets were used for all channel models. Convergence of the $\sum P_g$ values with more and more message sequences, as discussed below, is taken as evidence that a sufficient number of patterns is being used in all cases.

The $\sum P_g$ algorithm was found to be very efficient, using an average of 15 seconds of Univac 1108 Computer CPU time to evaluate typical 32nd degree polynomials.

Table 2.2 presents results for evaluation of the $\sum P_g$ algorithm in several respects. The table is constructed to tabulate $\hat{P}_u$ defined by

$$\hat{P}_u = \frac{\tilde{P}_u(n)}{n\,P(1)}$$

for comparison to Johnson's [21] determination of this quantity for several codes, where the quantity $\tilde{P}_u(n)$ is computed from (2.21) .

Table 2.2 $\sum P_g$; Probability of Undetected Error, $\hat{P}_u$, for Various β and $\hat{P}_u$ from Johnson Method.

All Results for the Pareto Model

Notes	16th Degree Code Polynomials (Octal)	$\beta = 5 \times 10^{-5}$ 864 message sequences		$\beta = 1.0 \times 10^{-5}$ 6117 message sequences		$\beta = 2 \times 10^{-6}$ 32,362 message sequences		Johnson
		$\sum P_g$	$\hat{P}_u$	$\sum P_g$	$\hat{P}_u$	$\sum P_g$	$\hat{P}_u$	$\hat{P}_u$
CRC-16	100003	2.59×10^{-3}	2.93×10^{-5}	2.68×10^{-3}	3.03×10^{-5}	2.71×10^{-3}	3.06×10^{-5}	2.83×10^{-5}
CCITT	170037	3.81×10^{-5}	4.31×10^{-7}	4.03×10^{-5}	4.55×10^{-7}	4.55×10^{-5}	5.14×10^{-7}	4.1×10^{-7}
SDLC	117615	7.99×10^{-6}	9.03×10^{-8}	1.19×10^{-5}	1.34×10^{-7}	2.04×10^{-5}	2.31×10^{-7}	9.5×10^{-8}
A BCH	142631	5.77×10^{-4}	7.50×10^{-6}	1.19×10^{-3}	1.55×10^{-5}	1.52×10^{-3}	1.98×10^{-5}	
PRIMITIVE	160115	3.57×10^{-6}	4.64×10^{-8}	7.91×10^{-6}	1.03×10^{-7}	1.52×10^{-5}	1.98×10^{-7}	6.7×10^{-8}
FIRE	176053	4.00×10^{-6}	5.20×10^{-8}	6.45×10^{-6}	8.38×10^{-8}	1.07×10^{-5}	1.39×10^{-7}	6.1×10^{-8}
FIRE	101617	7.47×10^{-6}	9.71×10^{-8}	1.57×10^{-5}	2.04×10^{-7}	2.25×10^{-5}	2.92×10^{-7}	
FIRE	107713	8.73×10^{-6}	1.13×10^{-7}	1.19×10^{-5}	1.55×10^{-7}	1.58×10^{-5}	2.05×10^{-7}	
FIRE	165523	9.44×10^{-6}	1.23×10^{-7}	1.74×10^{-5}	2.26×10^{-7}	2.14×10^{-5}	2.78×10^{-7}	
NON-PRIM	150355	1.16×10^{-6}	1.51×10^{-8}	1.17×10^{-4}	1.52×10^{-6}	2.37×10^{-4}	3.08×10^{-6}	
NON-PRIM	154163	1.20×10^{-6}	1.56×10^{-8}	1.01×10^{-5}	1.31×10^{-7}	7.28×10^{-5}	9.46×10^{-7}	
NON-PRIM	151717	1.24×10^{-6}	1.61×10^{-8}	2.30×10^{-4}	2.99×10^{-6}	3.51×10^{-4}	4.56×10^{-6}	
PRIMITIVE	133231	1.39×10^{-6}	1.81×10^{-8}	6.28×10^{-6}	8.16×10^{-8}	1.09×10^{-5}	1.42×10^{-7}	
PRIMITIVE	121617	1.78×10^{-6}	2.31×10^{-8}	1.84×10^{-5}	2.39×10^{-7}	3.34×10^{-5}	4.34×10^{-7}	
PRIMITIVE	123735	1.81×10^{-6}	2.35×10^{-8}	1.10×10^{-5}	1.43×10^{-7}	1.70×10^{-5}	2.21×10^{-7}	
PRIMITIVE	111713	1.82×10^{-6}	2.37×10^{-8}	1.55×10^{-5}	2.02×10^{-7}	2.85×10^{-5}	3.70×10^{-7}	
PRIMITIVE	175043	1.82×10^{-6}	2.37×10^{-8}	7.61×10^{-6}	9.89×10^{-8}	2.14×10^{-5}	2.78×10^{-7}	

Table is constructed for a block size n = 3200, except for the CRC-16, CCITT and SDLC polynomials for which n = 4000; $\hat{P}_u = P_u/(n\, P\{1\})$

Note that Johnson's results for the CRC, CCITT and SDLC codes correspond closely to the $\sum P_g$ results for 864 message sequences. For the two other codes evaluated in the present study using his method, the results seem to fall between the 864 and the 6,117 message sequence data.

The data in Table 2.2 can also be used to form a judgement as to the rate of convergence of $\sum P_g$ to a limiting value as more and more message sequences are used. In this regard, note that for the CRC-16 polynomial, for which $\sum P_g$ is large, little change in $\sum P_g$ results from the change from 6,117 message sequences to 32,362 message sequences. For the polynomials with smaller $\sum P_g$, however, the results converge less rapidly with the number of message sequences. For the smallest $\sum P_g$ in the table, (that for polynomial 150555), the fractional increment for $\sum P_g$ between 864 and 6,117 message sequences is 101 whereas that between 6,117 and 32,362 message sequences is 2.02.

Rate of convergence was studied in more detail for a specific 32 degree polynomial and several channel models. The polynomial chosen had close to the smallest $\sum P_g$ for all channel models. The results presented in Figure 2.4 seem to indicate satisfactory convergence, and hence a good estimate of $P_u(n)$, for all channel models, including the Brayer Table 3 model.

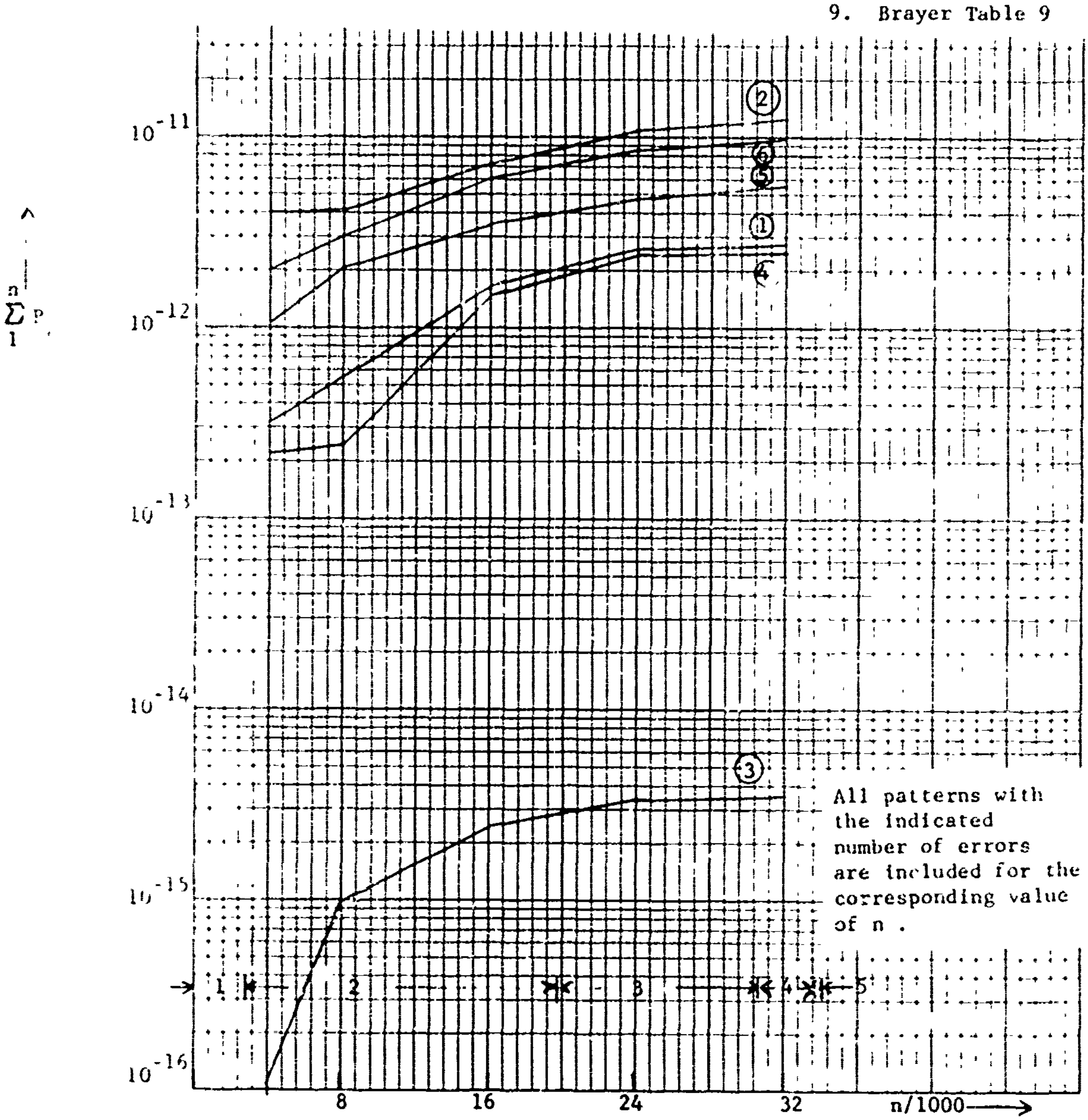
5. Results of Studies using the $\sum P_g$ Algorithm

Two extensive computer studies of classes of codes, as determined by generator polynomials, were carried out. All 900 irreducible 16th degree polynomials, as listed for example by Peterson [17], were evaluated using $\beta = 5 \times 10^{-5}$. The results are given in Table 2.3 along with the results for three good nonprimitive polynomials.

All 32nd, 31st and 30th degree irreducible polynomials listed in the Peterson tables were used to construct 32 check bit code polynomials,

Figure 2.4 $\sum_{k=1}^{n} P_g(k)$ versus n for the Recommended Polynomial (octal) 40460216667 and Several Channel Models

1. Pareto
2. Markov--Fritchman
3. Markov--Tasi
4. Brayer Table 3
5. Brayer Table 5
6. Brayer Table 6
7. Brayer Table 7
8. Brayer Table 8
9. Brayer Table 9



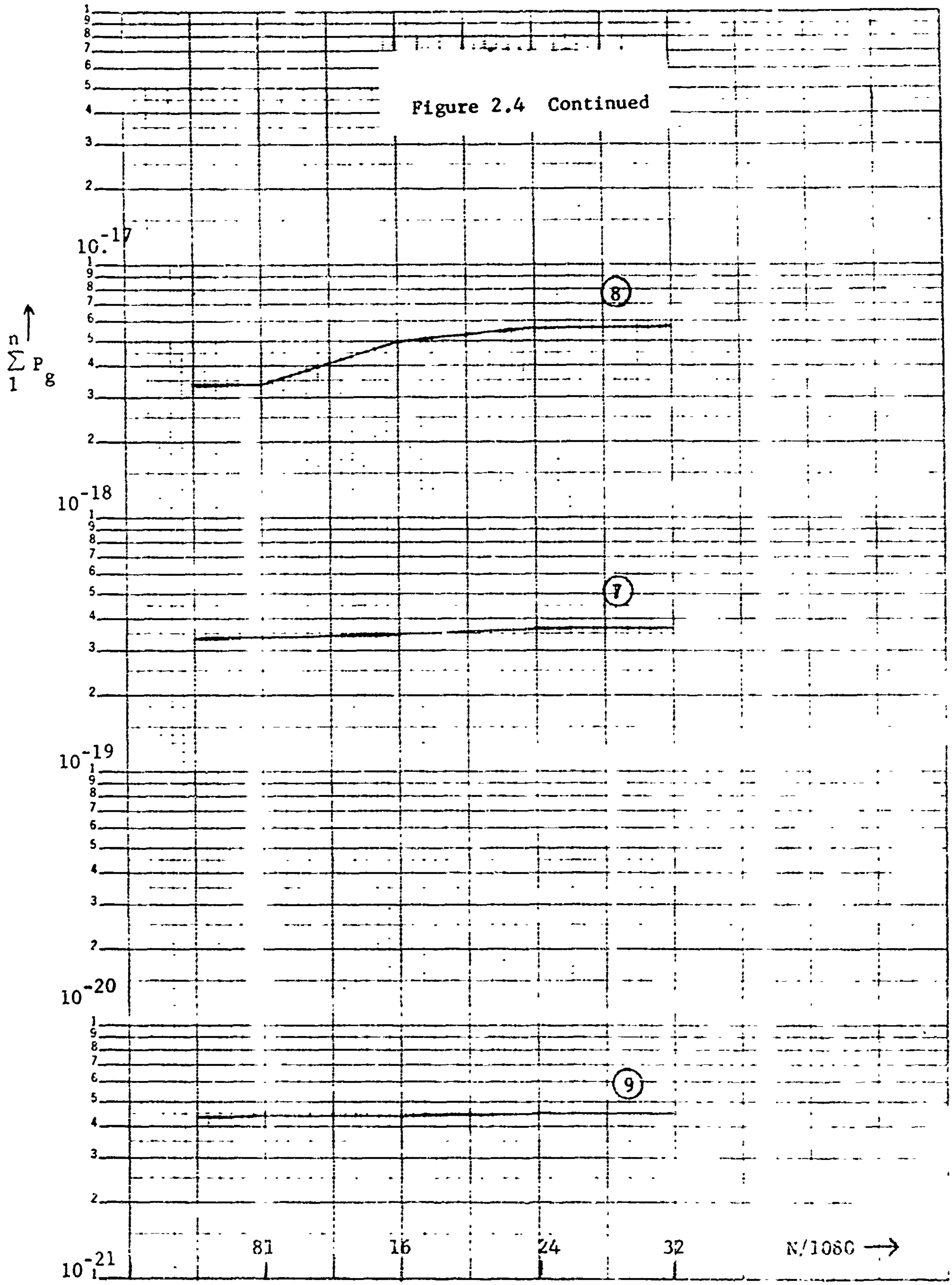


Figure 2.4 Continued

Table 2.3 $\sum P_g$ Figures of Merit for Selected 16th Degree Polynomials Calculated with $\beta = 5 \times 10^{-5}$ (864 information bit sequences)

Ranking	Polynomial (Octal)	$\sum P_g$	Notes
1	133231	.139323-05	Ranking with respect to irreducible polynomials
2	121617	.177967-05	
3	123735	.181136-05	
4	111713	.182194-05	
5	175043	.182496-05	
450	157315	.609053-03	
895	177775	.529636-03	
896	114011	.558423-03	
897	172621	.613355-03	
898	100201	.625100-03	
899	100021	.728813-03	
900	100003	.258801-02	
	150355	.116297-05	Best nonprimitive polynomials (limited search)
	154163	.119776-05	
	151717	.123954-05	

the 31st and 30 degree polynomials being multiplied by $1 + x$ and $1 + x^2$ respectively. The table lists 109 30th degree, 11 31st degree and 22 32nd degree polynomials. Codes based on each of these polynomials were investigated for the Pareto model. The $\sum P_g$ for the best ones in each group is tabulated in Table 2.4 for $\beta = 4.2 \times 10^{-6}$ and 8.7×10^{-7}. Numbers in parentheses in the first column indicate the rank for $\beta = 4.2 \times 10^{-6}$ within groups of the same degree polynomials. Similar numbers in the fourth column indicate the rank for $\beta = 8.7 \times 10^{-7}$ over the whole group of codes.

Table 2.4 $\sum P_g$ Values for "Good" 32nd Degree Polynomials for two Values of β using the Pareto Model

Polynomial Class	Polynomial (Octal)	$\sum P_g \times 10^{-12}$, $\beta = 4.2 \times 10^{-6}$	$\sum P_g \times 10^{-11}$, $\beta = 8.7 \times 10^{7}$
32 degree (1)	60537314115	.96	1.150 (6)
32 degree (2)	40460216667	1.00	.270 (1)
31 (1+X) (1)	60120240653	66.77	84.016 (7)
30 (1+X) (1)	52414670717	.601	.749 (4)
30 $(1+X^2)$ (2)	62613476131	.970	.291 (2)
30 $(1+X^2)$ (3)	51474633517	1.094	.459 (3)
30 $(1+X^2)$ (4)	54114300535	1.420	.815 (5)

Several classes of BCH-Fire codes were constructed to satisfy the requirement of 32 check bits and a block length of 2000 bits. Such codes have generator polynomials of the form [18]

$$g(X) = (X + 1)\,(X^{2\ell - 1} + 1)\, g_{BCH}(X) \, . \qquad (2.22)$$

As discussed in Part I of the report, the BCH polynomial, $g_{BCH}(X)$, can be expressed as

$$g_{BCH}(X) = \mathrm{LCM}\left[m_1(X), m_3(X), \ldots, m_{2t-1}(X)\right] \tag{2.23}$$

where LCM denotes least common multiple, and $m_i(X)$ is the minimum polynomial of α^i where α is a root of a primitive mth degree polynomial. For an effective code the block length, n, must satisfy

$$n = 2^m - 1 \geq 2000, \tag{2.24}$$

from which $m \geq 11$.

Since each $m_i(X)$ in (2.23) has degree m or less, the possible code classes of the form given in (2.22) which have 32 check bits and a block length $n \geq 2000$ are the six listed below:

1. $(X^{10} + 1)\, m_1^{(11)}(X)\, m_2^{(11)}(X)$
2. $(X^8 + 1)\, m_1^{(12)}(X)\, m_2^{(12)}(X)$
3. $(X^6 + 1)\, m_1^{(13)}(X)\, m_2^{(13)}(X)$
4. $(X^4 + 1)\, m_1^{(14)}(X)\, m_2^{(14)}(X)$
5. $(X^2 + 1)\, m_1^{(15)}(X)\, m_2^{(15)}(X)$
6. $(X^{2\ell} + 1)\, m_1^{(32 - 2\ell)}(X)$

ΣP_g was computed for all 104 codes of type 5 for $\beta = 4.2 \times 10^{-6}$ and 8.7×10^{-7}. The results, for the best and worst codes, are given in Table 2.5 with a ranking in parentheses in the first column for $\beta = 4.2 \times 10^{-6}$ and a similarly denoted ranking in the third column for $\beta = 8.7 \times 10^{-7}$.

Table 2.5 $\sum P_g$ Values for a Selection of Code Polynomials of the Form: $(X^2 + 1)\, m_1^{(15)}(X)\, m_2^{(15)}(X)$ for Two Values of β using the Pareto Model

Polynomial (Octal)		$\sum P_g \times 10^{-12}$, $\beta = 4.2 \times 10^{-6}$ (6117 message sequence)	$\sum P_g \times 10^{-11}$, $\beta = 8.7 \times 10^{-7}$ (32,362 message sequence)	
47665475341	(1)	.636	.380	(5)
56111263425	(2)	.718	.260	(3)
72450733617	(3)	.766	.251	(2)
54766326031	(4)	.858	.185	(1)
53760445455	(5)	1.188	.880	(8)
43611250751	(6)	1.365	.353	(4)
67007252603	(7)	1.441	.849	(7)
70425300155	(8)	1.473	.480	(6)
42323255113	(100)	87.03		
53614073271	(101)	101.67		
76577327771	(102)	124.22		
74467714763	(103)	222.11		
51224036761	(104)	3348.39		

With reference again to the six classes of codes given on page 66 the following classes were also exhaustively studied for $\beta = 8.7 \times 10^{-7}$ (32, 362 message sequences) using ten channel models :

codes of class 6 for ℓ = 0, 1 and 2 for available tabulated polynomials

all codes of class 5

all codes of class 3 (4 channel models)

polynomials 75626604261 and 40050004005 suggested by Brayer and McKee[†]

A summary of the results for the best polynomials is given in Table 6.

---

† Personal Correspondence

Table 6. ΣP_g for the Best 32nd Degree Generating Polynomials Using 32,362 Message Sequences

Polynomials (Octal)	Pareto ΣP_g 10^{-11}	Rank	Tsai ΣP_g 10^{-11}	Rank	Fritchman ΣP_g 10^{-11}	Rank	Brayer Table 3 ΣP_g 10^{-13}	Rank	Brayer Table 4 ΣP_g 10^{-12}	Rank	Brayer Table 5 ΣP_g 10^{-10}		Brayer Table 6 ΣP_g 10^{-10}	Rank	Brayer Table 7 ΣP_g 10^{-18}	Rank	Brayer Table 8 ΣP_g 10^{-17}	Rank	Brayer Table 9 ΣP_g 10^{-20}	Rank
Class 6																				
40460216667	.270	1	.271	1	1.250	3	.034	1	2.36	4	.0543	1	.097	1	.371	1	.567	1	.489	2
62613476131	.291	2	.736	2	1.572	4	.156	2	2.22	3	.103	3	.219	4	19.9	5	76.7	5	3.09	5
51474633517	.459	3	1.333	4	1.058	1	.280	3	1.23	1	.189	6	.352	6	22.2	6	119.6	6	3.16	6
52414670717	.749	4	1.316	3	1.172	2	.814	5	1.77	2	.161	5	.219	5	1.23	4	6.73	4	.716	4
54114300535	.815	5	2.475	5	2.317	5	.808	4	3.75	6	.0781	2	.114	2	.551	2	.977	2	.482	1
60532314115	1.150	6	4.635	6	3.771	6	2.16	6	3.03	5	.126	4	.200	3	.938	3	2.07	3	.656	3
60170240653	8.402	7	9.160	7	13.80	8	14.6	7	41.5	7	.718	7	1.26	7	127.9	7	676.0	7	7.52	7
Class 5																				
54766326031	.185	1	.529	3	.987	2	.230	2	.820	1	.382	7	.457	7	4.23	4	28.1	5	1.42	3
72450733617	.251	2	.348	1	1.022	3	.166	1	2.64	7	.1114	3	.233	5	3.56	3	8.39	3	1.77	4
56111263425	.260	3	.721	4	1.179	4	.235	3	2.06	6	.153	6	.239	6	35.5	7	39.8	7	8.47	7
43411250251	.353	4	.396	2	.934	1	.747	5	1.51	3	.996	8	1.08	8	6.89	5	34.8	6	2.36	5
47665475241	.380	5	.923	6	2.018	6	1.12	6	1.73	5	.126	4	.142	1	26.9	6	148.3	8	11.5	8
70425300155	.480	6	.825	5	1.641	5	.351	4	1.62	4	.111	2	.193	4	.134	1	.438	1	.210	2
67007252603	.849	7	8.247	8	2.358	7	5.11	8	.949	2	.108	1	.152	2	.171	2	1.67	2	.141	1
53760445455	.880	8	1.335	7	2.952	8	2.14	7	13.15	8	.141	5	.177	3	50.1	8	27.2	4	6.88	6
Class 3																				
74626213177							.0457	1												
46144043613							.0773	2	.855	5										
76463567721							.0849	3	1.01	8										
54726657475							.0951	4	.759	2	.0924	8								
51554074201							.10	5			.0566	9								
63430176277							.102	6			.0796	4	.133	5						
41352404643							.109	7												
74344725321							.112	8	.756	1	.0970	10								
42370706413							.116	9			.0548	1	.102	1						
44571017745							.136	10												
57162217443									.807	3	.0907	7	.143	9						
43326051501									.839	4	.0682	3	.114	2						
70[illegible]3223575									.915	6										
62244269743									.927	7	.0892	6	.118	3						
63379[illegible]75577									1.08	9										
[illegible]									1.22	10										
[illegible]6222[illegible]165											.0610	2	.122	4						
52710607323											.0883	5								
67504753537													.1331	6						
44072424733													.137	7						
57574602541													.140	8						
710[illegible]7721311													.146	10						
75520605261	2.14		19.31		15.4		3.96		7.51		.528		.508		.778		5.77		.413	
40050004005	7,775		12,790		1388		51818		18355		686.1		830.6		1.84×10^9		3.31×10^8		1.5×10^{10}	

COMMENTS

The 1st 7 poly. in the Pareto ranking are listed for class 6 & similarly the 1st 8 for class 5. For class 3 the 1st 10 poly. for each of 3 models are listed.

Rank: Class 6 is best of all 32, 31, and 30 degree poly. listed in Peterson times 1, $(1 + x)$ and $(1 + x^2)$ respectively. Class 5, best of 104 possibilities. Class 3, best of 103 possibilities.

6. Choice of a Code Polynomial

From the data presented in Section 5, it is clear that the $\sum P_g$ figure of merit varies over many orders of magnitude for the polynomials investigated. Furthermore, Table 6 shows that the figure of merit is sensitive, to some extent, to the channel model. On the other hand, the sensitivity to the channel model is not severe and a relatively large number of the codes considered in Table 6 could be considered essentially equivalent.

The best code polynomial in Table 6 for a particular channel model can be easily selected. For general use with the channel model unspecified, however, there seems to be no clear cut basis on which to choose between several polynomials which perform exceptionally well for some channel models and less well for others. For example, a good case can be made for the polynomials (octal) 40460216667, 54766326031, 70425300155, 42370206413 and 75626604261 as well as for several other polynomials.

To be specific, the polynomial (octal) 40460216667 or

$$g(X) = X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^{8} + X^{7} + X^{5} + X^{4} + X^{2} + X + 1 \qquad (2.25)$$

was chosen for recommendation and for further study.

Table 2.7 lists the following information for the recommended polynomial:

a) The rank of the polynomial for each of 10 models with respect to all 32nd degree polynomials evaluated

b) The figure of merit, $\sum P_g$

c) $P(1)$ - the probability of an error

d) $\bar{S}_p$

e) $\tilde{P}_u(n = 2000) = n\,\bar{S}_p\,P(1)\sum P_g$ - an estimate of P_u (2000)

Table 2.7 Characteristics of Recommended Polynomial

Channel Model	Recommended Polynomial (Octal)40460216667					Best Polynomial for each model	
	Rank *	ΣP_g	P(1)	$\tilde{S}_p$	$\tilde{P}_\mu$	ΣP_g	$\tilde{P}_\mu$
Pareto	4	2.7×10^{-12}	3.3×10^{-5}	1.75×10^{-2}	3.12×10^{-15}	1.85×10^{-12}	2.14×10^{-15}
Tsai	1	2.71×10^{-12}	1.8×10^{-3}	2.25×10^{-2}	2.2×10^{-13}		
Fritchman	7	1.25×10^{-11}	2.6×10^{-3}	2.0×10^{-2}	1.3×10^{-12}	9.34×10^{-12}	9.71×10^{-13}
Brayer Table 3	1	3.4×10^{-15}	3.4×10^{-5}	6.8×10^{-2}	1.57×10^{-17}		
4	20	2.36×10^{-12}	7.9×10^{-5}	8.0×10^{-2}	2.98×10^{-14}	7.56×10^{-13}	9.5×10^{-15}
5	1	5.43×10^{-12}	1.6×10^{-4}	6.8×10^{-3}	1.18×10^{-14}		
6	1	9.7×10^{-12}					
7	3	3.71×10^{-19}				1.34×10^{-19}	
8	2	5.67×10^{-18}				4.38×10^{-18}	
9	4	4.89×10^{-21}	$1.52 \times 10^{-}$ [illegible]	1.4×10^{-1}	2.08×10^{-22}	1.41×10^{-21}	6.0×10^{-23}

* For all polynomials studied.

The table also lists the figure of merit and estimated probability of undetected error for the best polynomial, (of those evaluated), for each channel model, if it is different from the recommended polynomial. The parameter $\overline{S}_p$ was not computed for channel models Brayer Table 6, 7 and 8 to conserve computer time. The values of $\overline{S}_p$ for these channel models are not expected to differ significantly from values for other models.

It can be noted from Table 2.7 that the recommended polynomial has an estimated probability of undetected error within a factor of approximately 3 of the best polynomial tailored to each channel model. The exact ratios of $\tilde{P}_u$ for the recommended polynomial to that of the best polynomial for each channel model are 1.45 (Pareto), 1.33 (Fritchman), 3.14 (Brayer Table 4) and 3.47 (Brayer Table 9). For four models the recommended polynomial is the best for the particular channel. For the two models for which $\overline{S}_p$ and hence $\tilde{P}_u$ was not computed, the ratio of $\sum P_g$ for the recommended polynomial to that of the best polynomial for the channel is 2.77 (Brayer Table 7) and 1.3 (Brayer Table 8).

As noted in Section 4, the curves of Figure 2.4 indicate the rate of convergence of the $\sum P_g$ algorithm for the recommended polynomial as more and more message sequences are used in the computation. Note that in most cases the change in $\sum P_g$ is almost negligible as the number of patterns is increased from 24,000 to 32,000.

As a final comment on the recommended polynomial, consider the following typical use. At a bit rate of 10^6 bits/sec, approximately 5×10^7 2000 bit patterns are transmitted per day. Interpreting probability as relative frequency, the largest estimated probability of error in Table 2.7, namely 1.3×10^{-12}, produces approximately one error on the average for every 10^{12} 2000 bit patterns. This occurs in 2×10^4 days or something like 50 years.

7. Conclusions for Part II

Part II of the report has dealt with the development, evaluation and application of an efficient algorithm for studying error detecting codes with respect to use on renewal channels.

With respect to the algorithm per se, it is efficient, using only tens of seconds of Univac 1108 CPU time on the average to compute the figure of merit for evaluating a polynomial, even for the largest collection of approximately 32,000 message patterns.

Even though the number of patterns for which probabilities are computed in evaluating $\sum P_g$ is a very small fraction of the total number of undetectable patterns, there is good evidence that $\sum P_g$ will change little through use of many more patterns. This evidence is provided by data on $\sum P_g$ as computed with more and more message patterns. The most extensive study of convergence, made for the recommended polynomial, shows an almost negligible change in $\sum P_g$ when the number of patterns is increased from 24,000 to 32,000 for all ten channel models.

Additional work done in an attempt to bound probability of undetected error and thus provide a further check on the accuracy of the algorithm did not give useful results. Bounds related to the BCH code were considered in detail in this part of the study. Generally speaking, typical bounds are too loose to be of significant value.

A further check on the accuracy of the algorithm is provided by the comparison with the work of Johnson [21] who estimates probability of undetected error using a different, although related, method. Agreement between the results of Johnson and those obtained with the $\sum P_g$ algorithm is good.

The $\sum P_g$ algorithm has been used to rank all 900 irreducible 16th degree polynomials with respect to the Pareto channel model.

For 32 check bit codes with block lengths of 2000 bits, it is shown that six classes of BCH-Fire codes encompass many of the commonly used types of codes. Three of these classes are investigated in detail in a study that considered a total of approximately 350 polynomials. There is no evidence to indicate that different results would be obtained from a study of the other three classes of BCH-Fire codes.

From this study it can be concluded that a group of possibly a dozen codes will provide the lowest undetectable error probability in general applications for which a precise channel model cannot be specified. The estimated probability of undetected errors for these "good" codes is on the order of 10^{-12}, a value which would produce one undetected error in something like fifty years at bit rates of 10^6 bits/second. Four polynomials were found to have undetected error probabilities as large as four or more orders of magnitude greater than those for good polynomials.

The code polynomial, $X^{32}+ X^{26}+ X^{23}+ X^{22} + X^{16}+ X^{12}+ X^{11}+ X^{10}+ X^8 + X^7 + X^5+ X^4+ X^2+ X + 1$, is recommended as a specific choice. The characteristics of this polynomial are investigated in detail and it is shown that the polynomial has a probability of undetected error no larger than on the order of three times that of the best polynomial tailored to each specific channel model. For four of the channel models considered this polynomial is the best of those considered.

PART III

GENERALIZATIONS

1. The Chien-Haddad Renewal Model: Results for a Special Case

The work reported in Part II of this report all uses renewal channel models which depend on first order gap statistics and hence are straightforward to identify. As discussed above, renewal models have been matched to a variety of practical channels; however, it is clear that not all channels can be modeled as renewal channels. The Chien-Haddad model is one of the most general nonrenewal model which has been considered in the literature.

Because of the complexity of nonrenewal models, such as the Chien-Haddad, the properties of such models are less clearly understood than the properties of renewal models and furthermore there is less agreement as to appropriate choices of parameters to match practical channels. As a part of the present study, two Chien-Haddad models were investigated and the results are compared to corresponding results for renewal models.

For simplicity a Chien-Haddad model using two by two matrices was assumed (corresponding to four elementary states). Reference to the discussion of the Chien-Haddad model in Part I indicates that to define such a model requires specification of π_1, π_2, q_1, q_2 and a 2 x 2 matrix T.

Since typical parameter values were not available, a somewhat arbitrary choice was made for the first model as noted below:

Chien-Haddad Model B

$$\pi = \lfloor 0.57143,\ 0.42857 \rfloor$$

$$q = \lfloor 10^{-5},\ 0.3 \rfloor$$

$$T = \begin{bmatrix} 0.7 & 0.3 \\ 0.4 & 0.6 \end{bmatrix}.$$

For this model P(m,n) was computed through use of (1.62) and (1.63) and the resulting curves are given in Figures A.21 and A.22 of Appendix I.

In order to compare the Chien-Haddad model to the renewal models studied, undetectable error patterns of small weight and length for several particular codes were determined and the probability of these patterns were determined for the Chien-Haddad model. (In the calculation it was necessary to use (1.55), which gives F(m + 1) for the Chien-Haddad model.)

For the parameters chosen for the Chien-Haddad model A the probability of typical error patterns was smaller by 17 orders of magnitude than corresponding probabilities for Pareto, Fritchman, or Tsai models. A model, termed Chien-Haddad Model B was constructed by adjusting the parameters so that typical error patterns for a fixed code had probabilities on the same order of magnitude as those for the renewal models. This resulted in the model specified below:

Chien-Haddad Model A

$$\pi = \left[0.985,\ 0.015\right]$$

$$q = \left[10^{-5},\ 0.3\right]$$

$$T = \begin{bmatrix} 0.999 & 0.001 \\ 0.066667 & 0.93333 \end{bmatrix}$$

Curves of P(m, n) for Chien-Haddad model B are given in Figure A.23 and A.24 of Appendix I.

Table 3.1 compares the total probability of a collection of error patterns for several 16th degree generating polynomials for the Chien-Haddad, the Pareto, the Fritchman, and the Tsai models. In each case the collection of undetectable error patterns is comparable to that used by Johnson [21] and to that resulting from $\delta = 5.8 \times 10^{-5}$ in the method discussed in Part II of the report.

Notes	Polynomial (Octal)	Pareto	Fritchman	Tsai	Chien-Haddad	
					B	A
SDLC	117615	8.8×10^{-8}	1.5×10^{-8}	1.7×10^{-8}	2.5×10^{-7}	1.0×10^{-25}
PRIMITIVE	160113	5.7×10^{-8}	5.8×10^{-9}	2.3×10^{-8}	9.7×10^{-8}	3.7×10^{-25}
FIRE	176053	5.[illegible] $\times 10^{-8}$	2.6×10^{-9}	8.5×10^{-9}	4.4×10^{-8}	1.1×10^{-24}
BCH	142631	—	—	—	4.0×10^{-6}	1.2×10^{-23}
CCITT	170037	—	—	—	2.5×10^{-6}	7.5×10^{-25}

Table 3.1 Comparison of Estimated Undetected Error Probability for 15th Degree Generating Polynomials and Various Channel Models, for n = 4000.

The probabilities of typical undetectable error patterns for the generator polynomial (Octal) 176053 were compared for the Pareto and the Chien-Haddad model B. It was observed that the probabilities of individual patterns did not correspond closely for the two models. For example for the group of patterns examined the largest probability using the Pareto model was a pattern of length 42 and weight 4 while the largest probability using the Chien-Haddad model B occurred for a pattern of weight 6 and length 23. A similar discrepancy was found for the smallest probability pattern.

The sensitivity of the Chien-Haddad model to a parameter in the T matrix was investigated by determining B(b, N)/B(0, N, as a function of p for the model specified below:

$$\pi = \left[\frac{p}{p + 0.0001} , \frac{0.0001}{p + 0.0001} \right]$$

$$q = \left[0.001 , 0.30 \right]$$

$$T = \begin{bmatrix} 0.9999 & 0.0001 \\ p & 1 - p \end{bmatrix}$$

The results given in Figure A.25 show an extreme sensitivity to p. Choices of p on the order of 0.1 would seem to correspond to practical channels which would not seem to strongly favor bursts of a particular length.

2. Approaches to Developing Channel Models Based on Physical Parameters.

As noted in the literature review of Part I, most existing channel models used in evaluating codes were developed by matching certain statistical properties of binary random sequences generated by some class of mathematical models to those of experimentally measured error sequences. This approach does not place in evidence the effect on the mathematical model of charging physical parameters such as signal-to-noise ratio or intersymbol interference, nor does it account directly for different types of modems.

The literature contains a variety of analog channel models, on the other hand, which are parameterized with physical variables. These models, however, have not been used with particular modem types to compute the statistics of digital error sequences.

In principle it is feasible to combine analog channel models with models of typical modems and then compute the statistics of appropriate digital error sequences. Such an approach gives the statistical representation for error sequences necessary for designing codes and also retains the parameterization in terms of physical parameters.

A step in this direction is taken in the present study by considering a very tractable channel/modem model for a binary differential phase shift keying system. An analysis of such a model by Salz and Saltzberg [30] is used as a starting point.

The system considered uses a modem modeled as consisting of a transmitter which generates an ideal waveform and a receiver consisting of an ideal input and output filter, an ideal delay, a sampler, and an optimum decision rule. The channel is represented simply as adding Gaussian noise which is statistically independent of the message process. Although the channel/modem models have no memory, use of the differentially coherent detector introduces a mechanism for memory over one past bit and thus the system has a reasonable probability of double error.

Salz and Saltzberg derive expressions, (equations (19) and (21), p. 204 of [30]), for the probability, P(1, 1), of double errors and the conditional probability, P(1|1), of an error given that an error has occurred. The results are

$$P(1,1) = \frac{1}{4\pi} e^{-M} \int_0^{\pi} \{[1 - \mathrm{erf}(\sqrt{M} \cos \theta)]^2$$

$$+ \sqrt{M\pi} \cos \theta \, e^{M \cos^2 \theta} [1 - \mathrm{erf}(\sqrt{M} \cos \theta)]^2$$

$$[1 + \mathrm{erf}(\sqrt{M} \cos \theta)]\} \, d\theta \qquad (3.1)$$

$$P(1|1) = \frac{P(1, 1)}{P(1)} = 2 e^{M} P(1, 1) \qquad (3.2)$$

where

$$M = 1/2\sigma^2 = \text{signal-to-noise ratio}$$

$$P(1) = \frac{1}{2} e^{-M}$$

$$\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt.$$

The expression for P(1, 1) can be evaluated numerically for a given value of M. Note that the assumptions of the model limit the memory to one past bit so that, for example, P(1|1,1) = P(1|1).

To study codes, it is desirable to evaluate the statistics of an error sequence. It will be shown, first, that the model specifies a renewal process, which is completely described by the gap probability, p(n), and the probability, P(1), of an error. The gap probability, p(n), will then be obtained.

For a renewal process the gap lengths are independent. Thus if $P(g_1, g_2, \ldots, g_k)$ is the probability of successive gaps of length $g_1, g_2, \ldots, g_k$, for a renewal process

$$P(g_1, g_2, \ldots, g_k) = p(g_1) \ldots p(g_k) \qquad (3.3)$$

Consider the joint probability of two gaps, $P(g_1, g_2)$, which can be expressed as

$$P(g_1, g_2) = P(0^{g_1-1}\,10^{g_2-1}\,1|1) \tag{3.4}$$

using the notation of Part II. Using the well known relations for conditional probability the following equalities result

$$P(0^{g_1-1}\,10^{g_2-1}\,1|1) = P(10^{g_1-1}\,10^{g_2-1}\,1)/P(1)$$

$$= P(0^{g_2-1}\,1|10^{g_1-1}\,1)\,\frac{P(10^{g_1-1}\,1)}{P(1)} = P(0^{g_2-1}\,1|10^{g_1-1}\,1)\;P(0^{g_1-1}\,1|1).$$

Using the fact that $P(0^{g_2-1}\,1|10^{g_1-1}\,1) = P(0^{g_2-1}\,1|1)$, and the definiton of $p(g)$ results finally in the equation

$$P(g_1, g_2) = p(g_1)\;p(g_2), \tag{3.5}$$

which shows that the model is a renewal process.

Further manipulation based on relations for conditional probability and the fact that memory extends only over one past bit yields an expression for $p(n)$, namely

$$p(n) = \frac{[P(0, 1)]^2\,[P(0, 0)]^{n-2}}{[P(0)]^{n-1}\;P(1)}\;. \tag{3.6}$$

The gap probability can be determined from the relations for $P(1)$ and $P(1, 1)$ using the following identities

$$P(1, 0) = P(0, 1) = P(1) - P(1, 1) \tag{3.7}$$

$$P(0, 0) = P(0) - P(0, 1) \tag{3.8}$$

Curves of p(n) versus n for various signal-to-noise ratios were computed and the results are given in Figure 3.1. The figure also gives the p(n) curves for several of the Brayer models. Note that the 8 db signal-to-noise ratio curve for the double error model very closely matches the Brayer Table 8 model. The data also suggests that several other Brayer models could be matched with appropriate signal-to-noise ratios.

3. Approaches to the Approximation of Nonrenewal Models with Renewal Models

The tractable algorithm for estimating the probability of undetectable error for specific codes is developed in Part II for renewal channels. There is a reasonable expectation that a similar, more complicated, algorithm can be developed for more general nonrenewal channel models.

An alternate approach to studying codes for nonrenewal channels, which is worth exploring, is to approximate the nonrenewal channel model with a renewal model which is equivalent in some sense. This section of the report suggests an approach which might be used.

A class of Markov processes, termed "unifilar Markov processes," which have useful approximation properties are defined and discussed in the literature of information theory, see for example Ash [30]. A Markov Chain is said to be unifilar with respect to the function φ if for each state C_k the states C_{k1}, C_{k2}, ... which can be reached in one step from C_k are such that $\varphi(C_{k1})$, $\varphi(C_{k2})$ ... are distinct values.

A subset of unifilar processes can be constrained to be renewal processes, although the details of the necessary constraints have not been worked out. In using the unifiliar process to represent the error properties of a channel, the function $\varphi(C_k)$ would be set equal to 0 (no error) for some states and 1 (error) for other states.

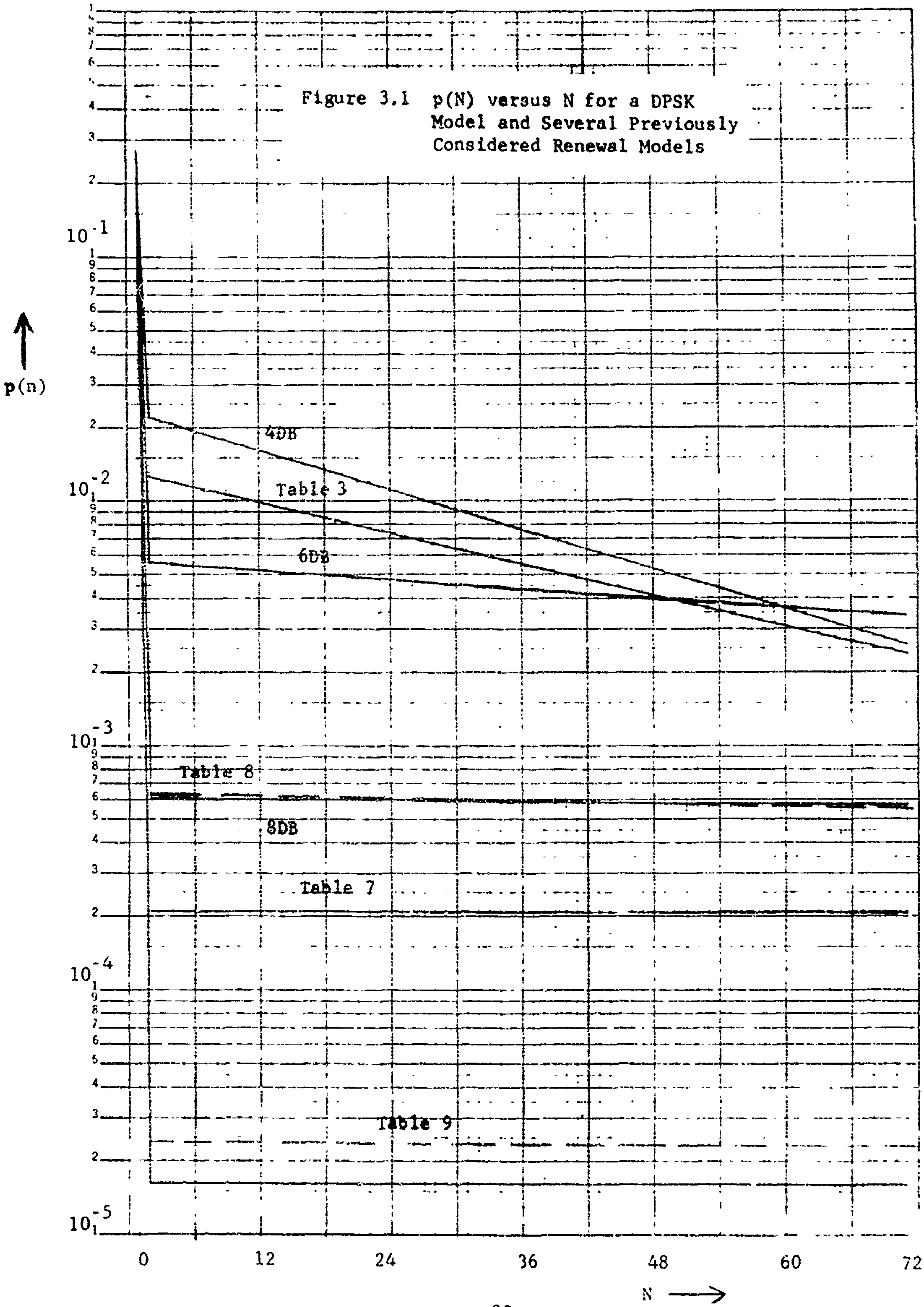


Figure 3.1 p(N) versus N for a DPSK Model and Several Previously Considered Renewal Models
p(n)
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{-5}
4DB
Table 3
6DB
Table 8
8DB
Table 7
Table 9
0
12
24
36
48
60
72
N

Unifiliar processes have the useful property that they can be used to approximate any other given Markov process of finite order in the sense of matching the "uncertainty" of the process arbitrarily closely. The uncertainty $H(x)$ of a process $\{x_i\}$ is defined as

$$H(x) = \lim_{n \to \infty} H(x_n | x_1, x_2, \ldots x_{n-1}) \tag{3.9}$$

where $H(x_n | x_1, x_2, \ldots x_{n-1})$ is the conditional uncertainty, or conditional entropy, of the sequence $\{x_1, \ldots x_n\}$, (see Ash [30], for example, for a definition of entropy).

The attractiveness of using entropy in generating an approximation is supported by two observations, namely:

1. If the function, $H(x_n | x_1, x_2, \ldots x_{n-1})$, is the same for two processes, then the nth order statistics of the processes are the same, and
2. The capacity of the channel is closely related to the entropy of the error sequence. Channel capacity is a natural parameter to use in describing a communication channel.

The order of a unifilar Markov process is defined as the minimum number of past values required to specify the current value in the sequence. Thus the order of a unifilar process $\{x_i\}$ required to approximate a process $\{y_i\}$ within an uncertainty error, ϵ, can be determined by requiring that

$$H(x_{n+1} | x_1, \ldots x_n) - H(y_{m+1} | y_1, \ldots y_m) \leq \epsilon \tag{3.10}$$

for all m [illegible] n.

Details of matching nonrenewal processes with renewal unifilar process have not yet been worked out. Consideration of a simple example seems to indicate that the method is feasible, although a large number of states may be required of the unifilar process.

4. Conclusions for Part III

Studies with a simple Chien-Haddad channel model have shown that this model is quite sensitive to its parameter values. Relatively small changes have a pronounced effect on burst error probabilities and hence on the probability of undetectable error sequences for codes, if this model is used.

It is possible to adjust the parameters of the Chien-Haddad model investigated to give sequence probabilities on the same order of magnitude as those for other (renewal) models. If this is done, computation of the probability of a selection of undetectable error sequences shows that sequence probabilities can be quite different for the Chien-Haddad model from other models studied.

The work in Part III, Section 2, shows that it is possible, in a tractable case, to combine analog channel models with modem models and compute the statistics of error sequences for binary operation. It turned out that the simple DPSK system studied gives a renewal process for the error sequence which, for certain signal-to-noise ratios, closely matches that of several empirical models studied in Part II.

The comments in the final section of Part III outline an approach to approximating nonrenewal models with renewal models. The method seems feasible but has yet to be evaluated in other than a trivially simple case.

5. Recommendations for Future Work

The work undertaken in the present study might be regarded as a first step in the general problem of choosing and evaluating error detecting (and possibly also error correcting) codes for large scale networks.

An attempt was made in the present study to identify problems that are of significant importance yet at the same time could be solved in a reasonable length of time. This led to the concentration on renewal channel models which are both tractable and represent a major fraction of the useful channels. Since code selection is based on channel models, almost exclusive emphasis in the study was given to techniques for selecting codes for renewal channels.

Future work should be directed toward a study of more general channel models such as the Chien-Haddad and to code selection procedures for these models. In dealing with practical systems, especially if degraded operation is to be considered, it would be very desirable to have channel models in terms of measurable physical quantities such as signal-to-noise ratio. Future work is also required in this area.

Work with, and related to, the use of empirical data in code selection is also desirable. First of all, the question of a sufficient collection of statistics to completely characterize a channel with respect to the coding problem seems to be completely open. On another aspect of the problem, because the probability of undetected error for practical codes is so small (on the order of 10^{-13}), "brute force" processing of recorded error sequences to evaluate codes is essentially out of the question. Alternatives to the brute force approach need to be developed.

Finally, it seems likely that complete communication network designs will be evaluated to some extent through simulation. The Monte Carlo approach of directly processing simulated data sequences is a natural method to use. Such an approach, however, is limited by the same small error probabilities that plague the use of measured data. Some alternative, such as conditioning or being near an error or the use of amplified error rates, must be perfected in order to be able to simulate systems under typical operating conditions.

APPENDIX I: Typical Channel Characteristics

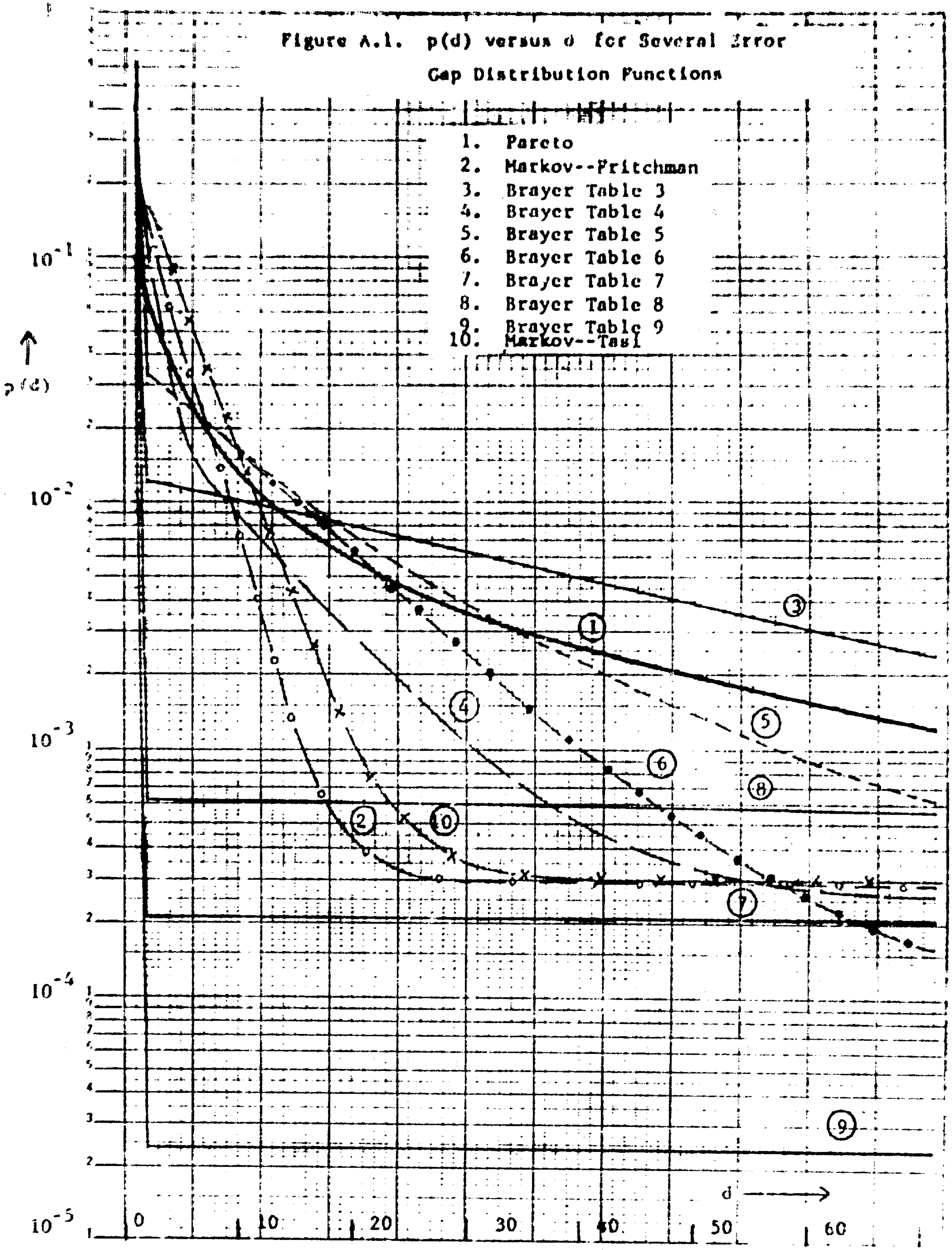
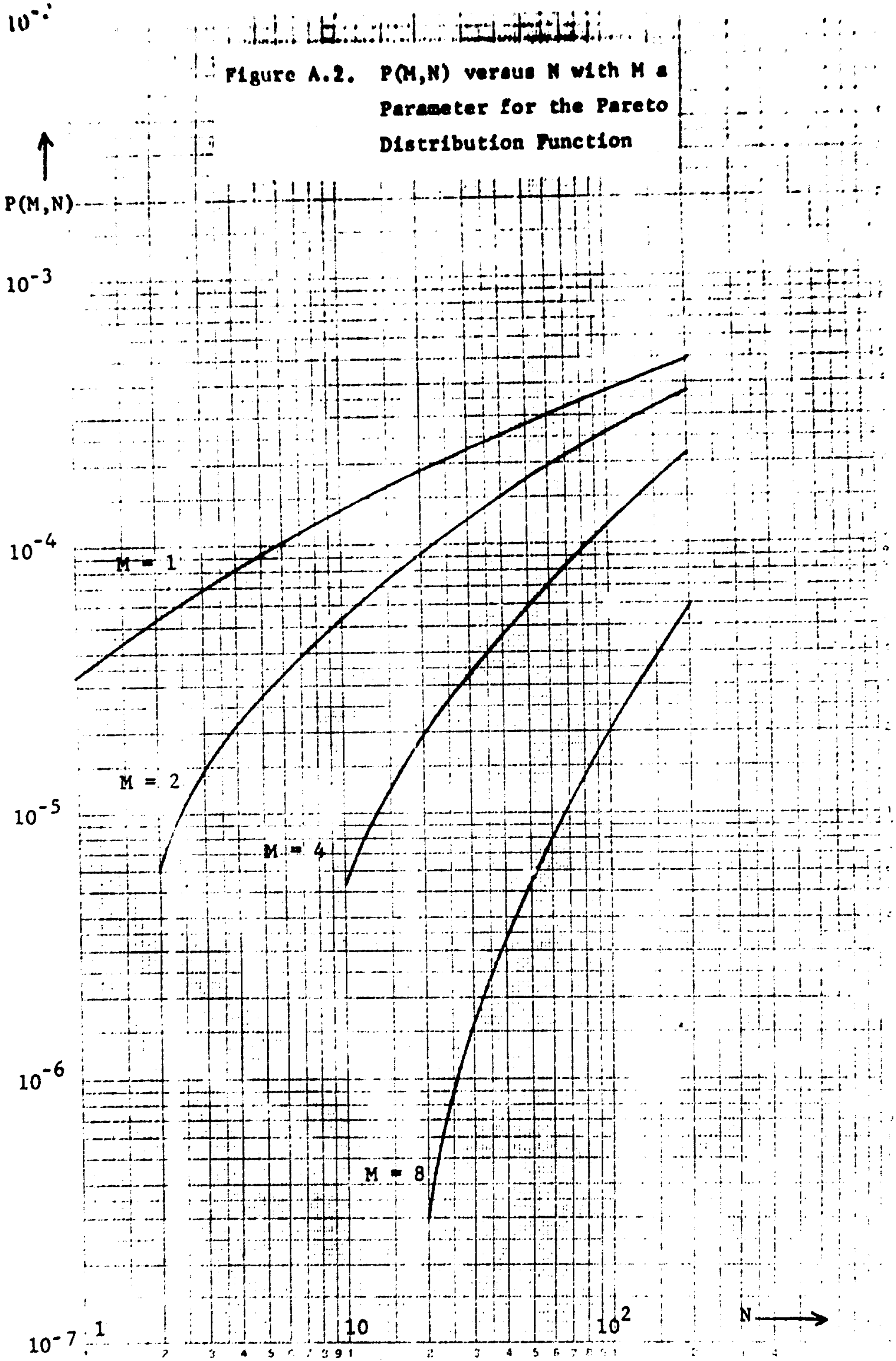


Figure A.1. p(d) versus d for Several Error
Gap Distribution Functions
1. Pareto
2. Markov--Fritchman
3. Brayer Table 3
4. Brayer Table 4
5. Brayer Table 5
6. Brayer Table 6
7. Brayer Table 7
8. Brayer Table 8
9. Brayer Table 9
10. Markov--Tsai
p(d)
d
10^-1
10^-2
10^-3
10^-4
10^-5
0
10
20
30
40
50
60

Figure A.2. P(M,N) versus N with M a Parameter for the Pareto Distribution Function



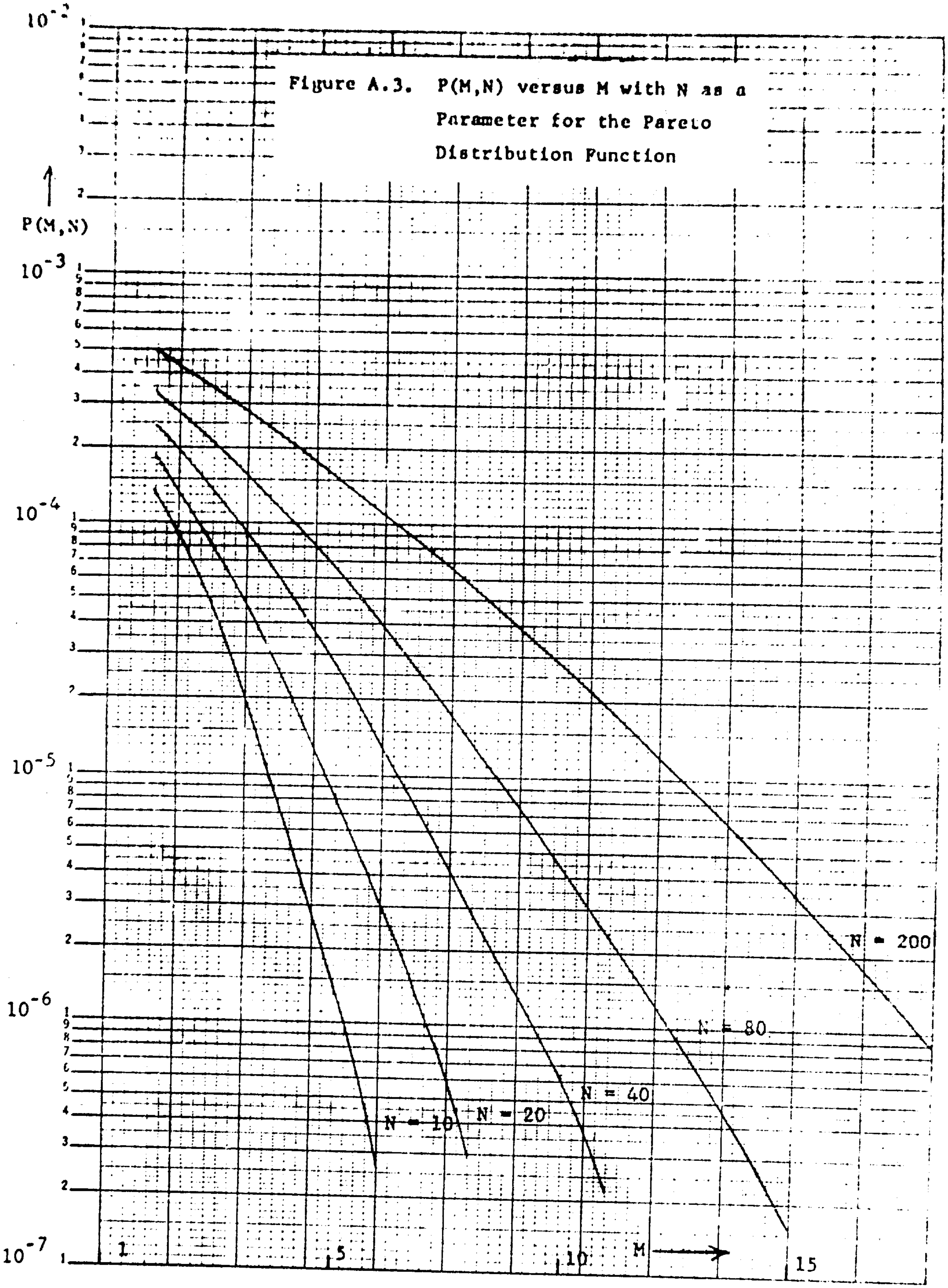


Figure A.3. P(M,N) versus M with N as a Parameter for the Pareto Distribution Function

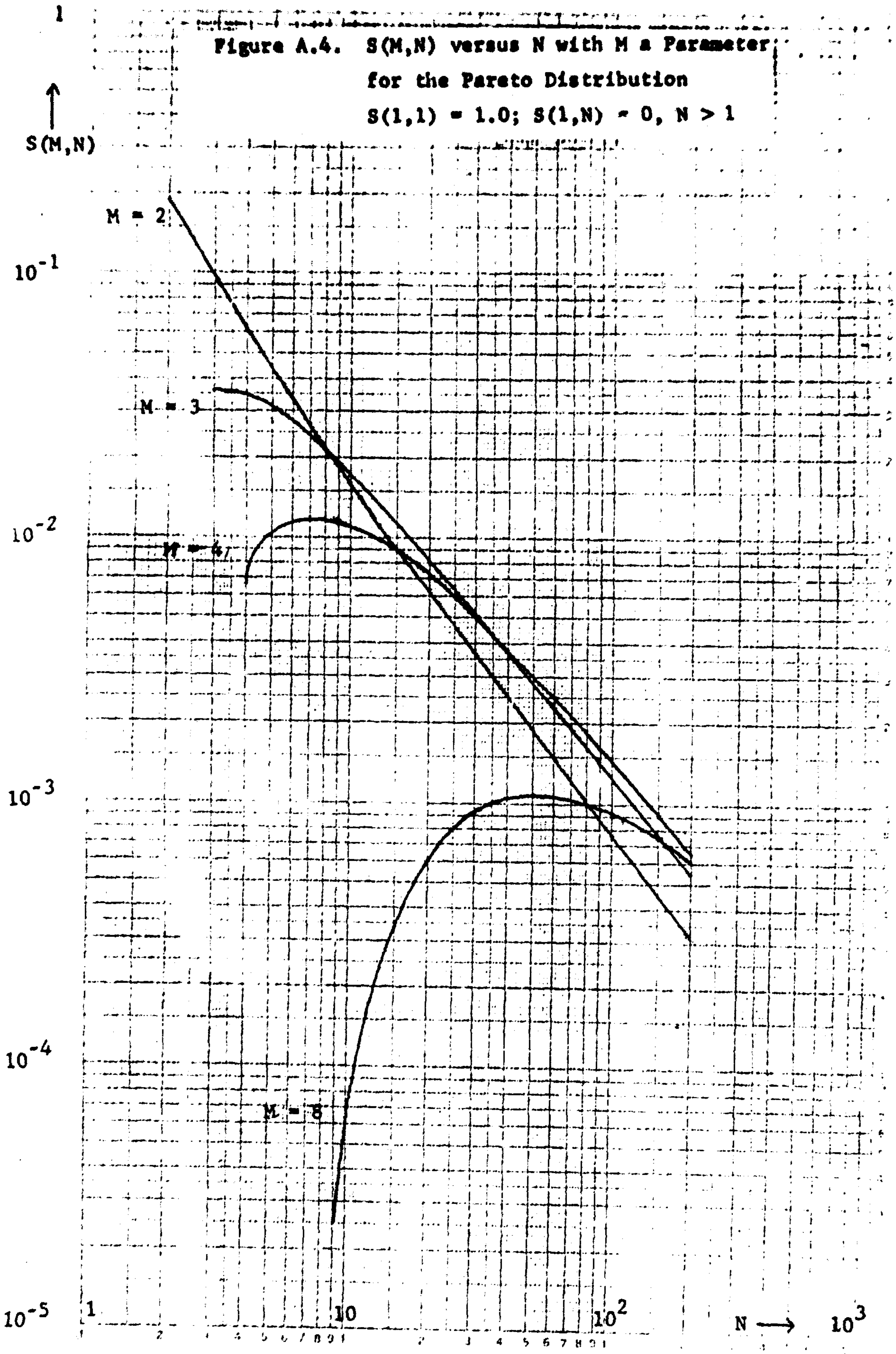


Figure A.4. S(M,N) versus N with M a Parameter for the Pareto Distribution
S(1,1) = 1.0; S(1,N) = 0, N > 1

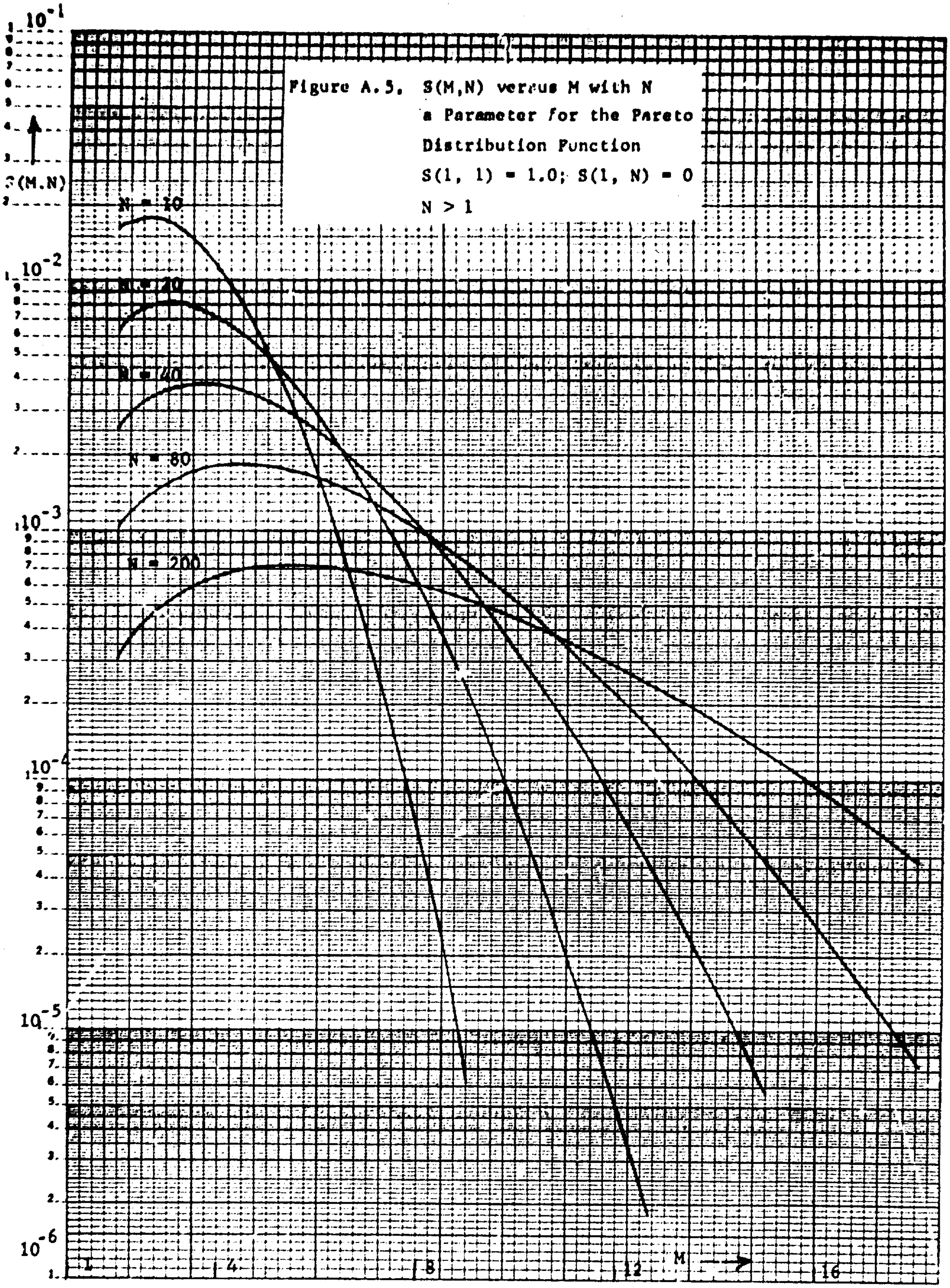
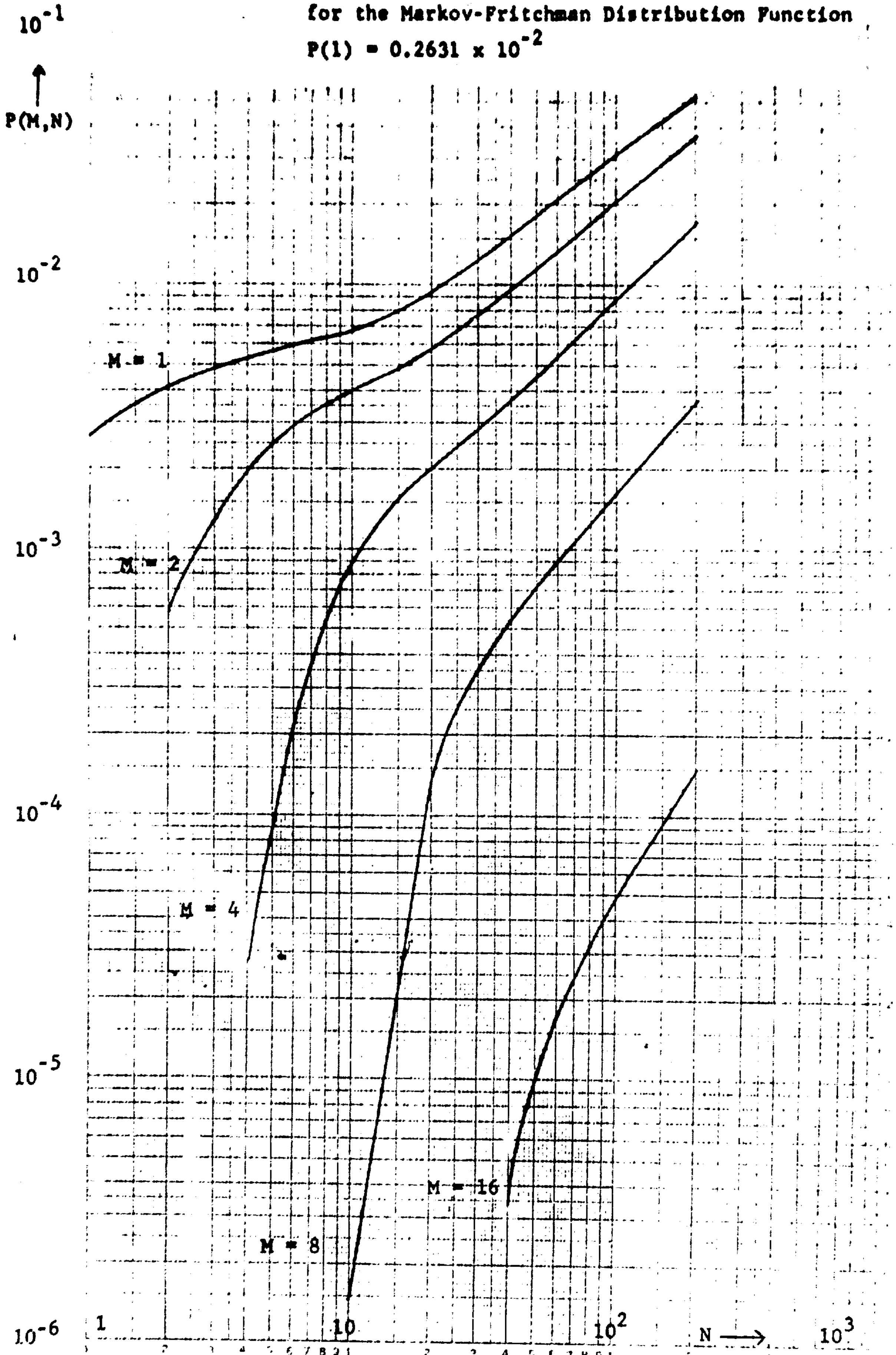


Figure A.5. S(M,N) versus M with N a Parameter for the Pareto Distribution Function
S(1, 1) = 1.0; S(1, N) = 0
N > 1

Figure A.6. P(M,N) Versus N with M a Parameter for the Markov-Fritchman Distribution Function $P(1) = 0.2631 \times 10^{-2}$



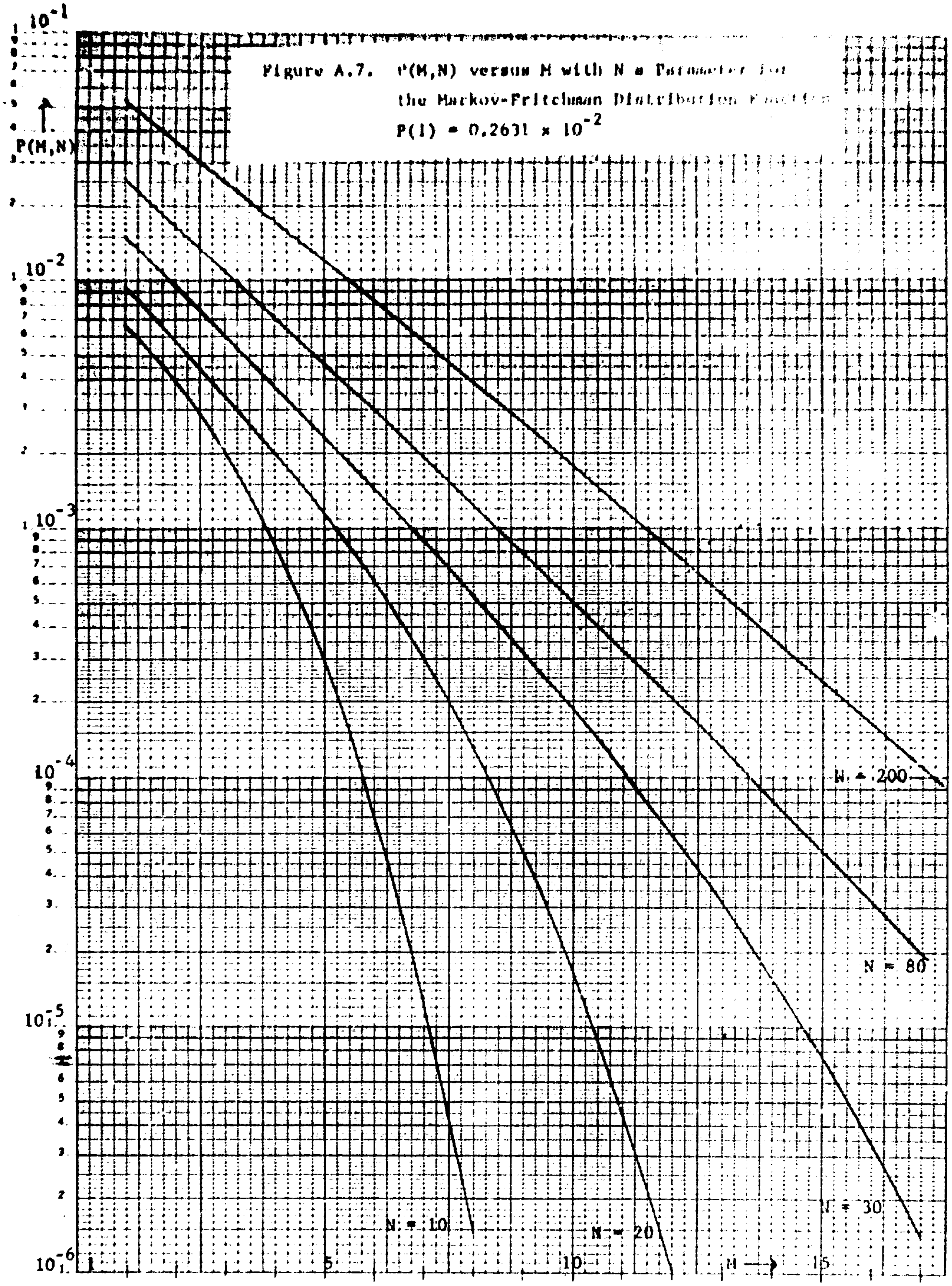


Figure A.7. P(M,N) versus M with N a Parameter for the Markov-Fritchman Distribution Function
$P(1) = 0.2631 \times 10^{-2}$

Figure A.8. S(M,N) versus N with M a Parameter
for the Markov-Fritchman Distribution Function
S(1,1) = 1.0; S(1,N) = 0 N > 1

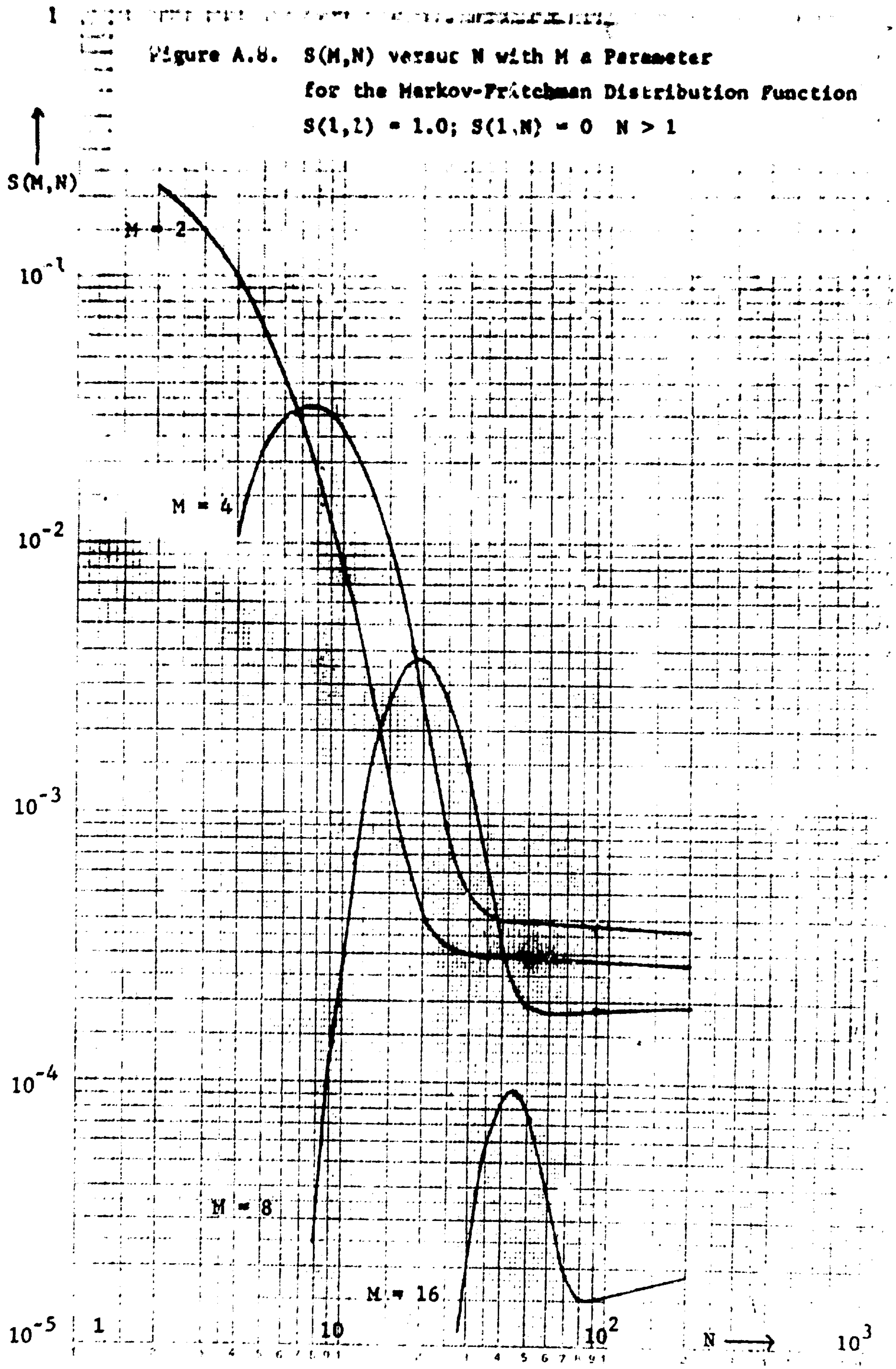


Figure A.9. S(M,N) versus M with N a Parameter
for the Markov-Pritchman Distribution Function
$S(1,1) = 1.0$; $S(1,N) = 0 \quad N > 1$

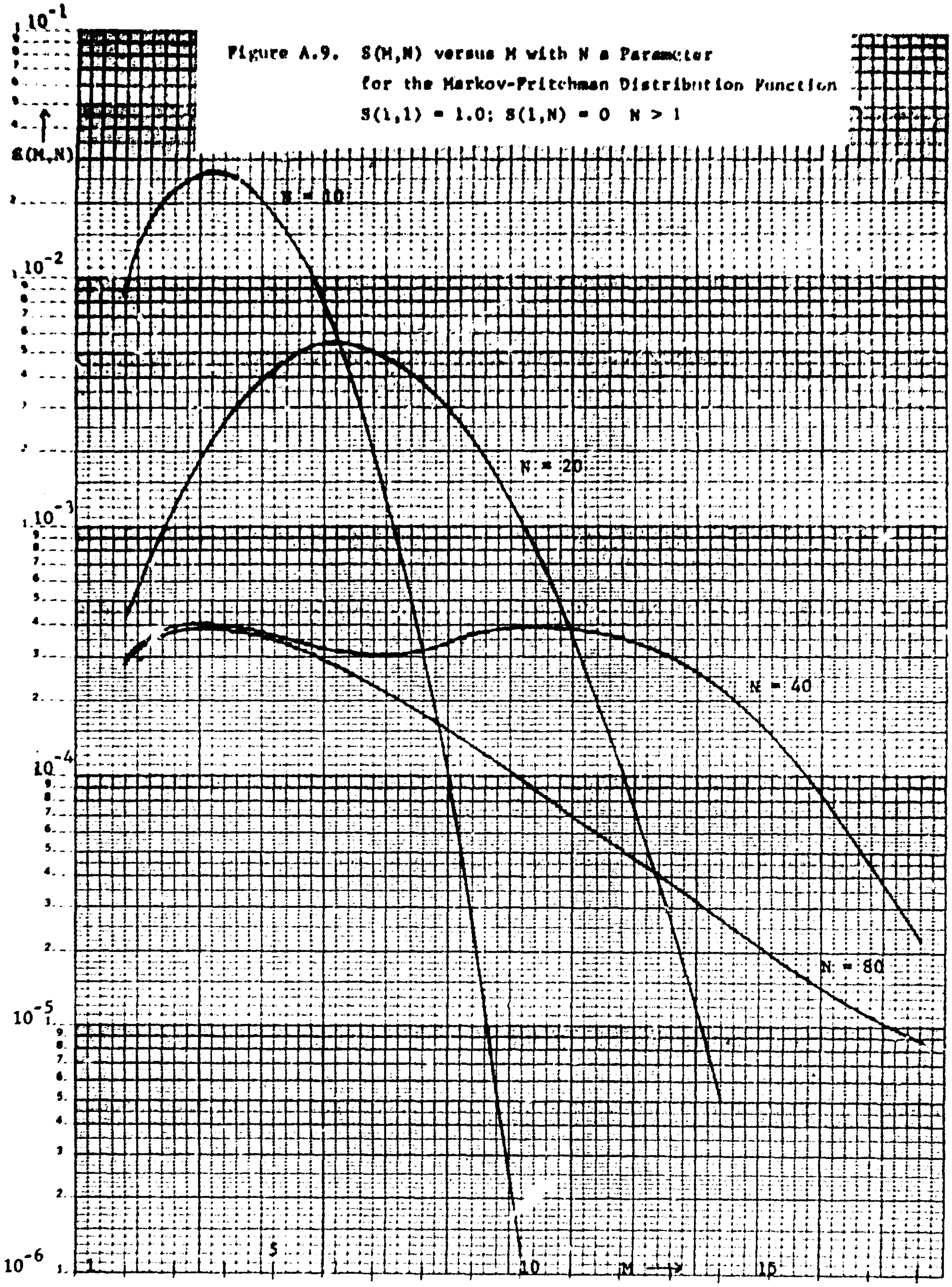
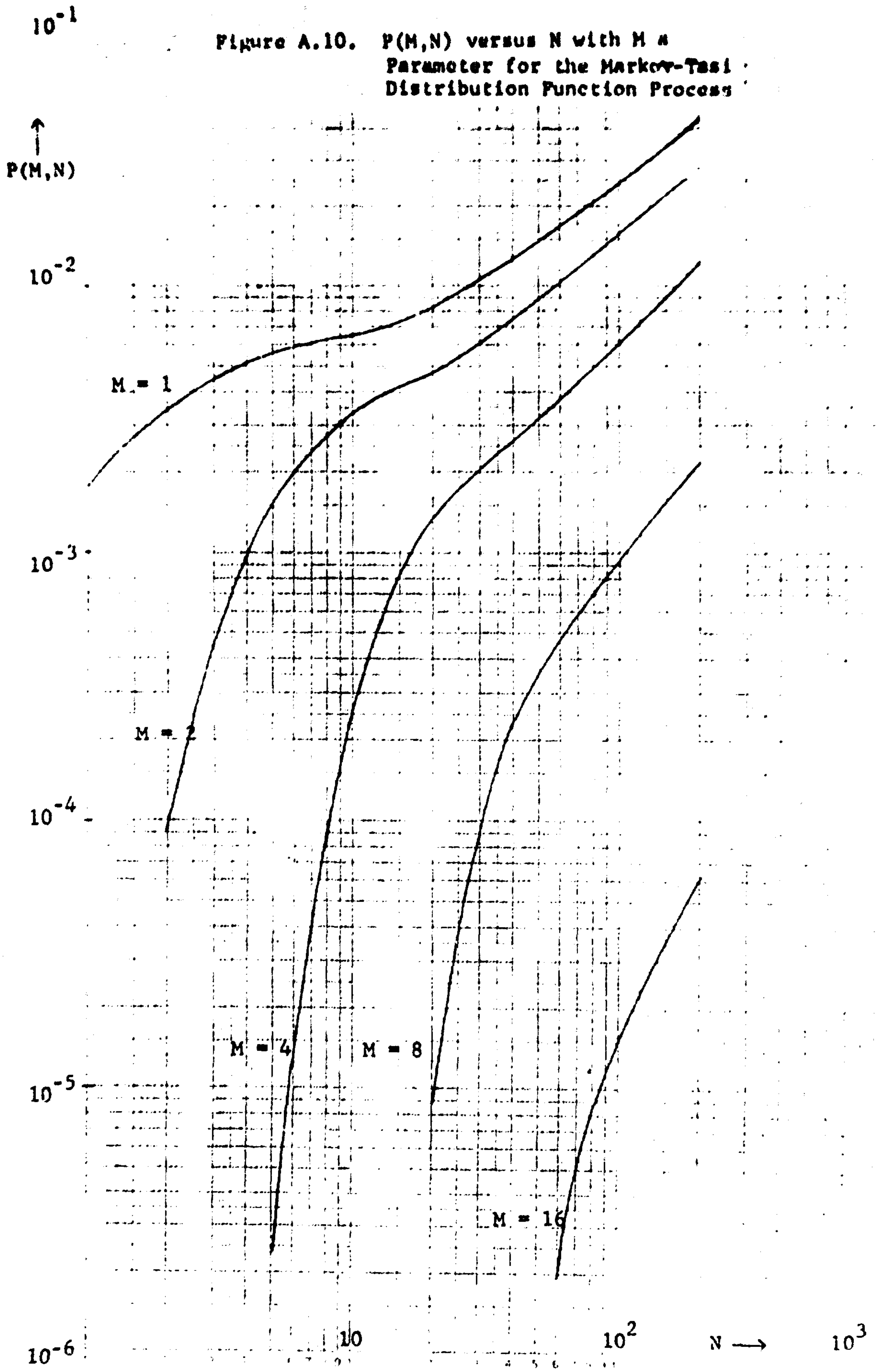


Figure A.10. P(M,N) versus N with M a Parameter for the Markov-Tasi Distribution Function Process



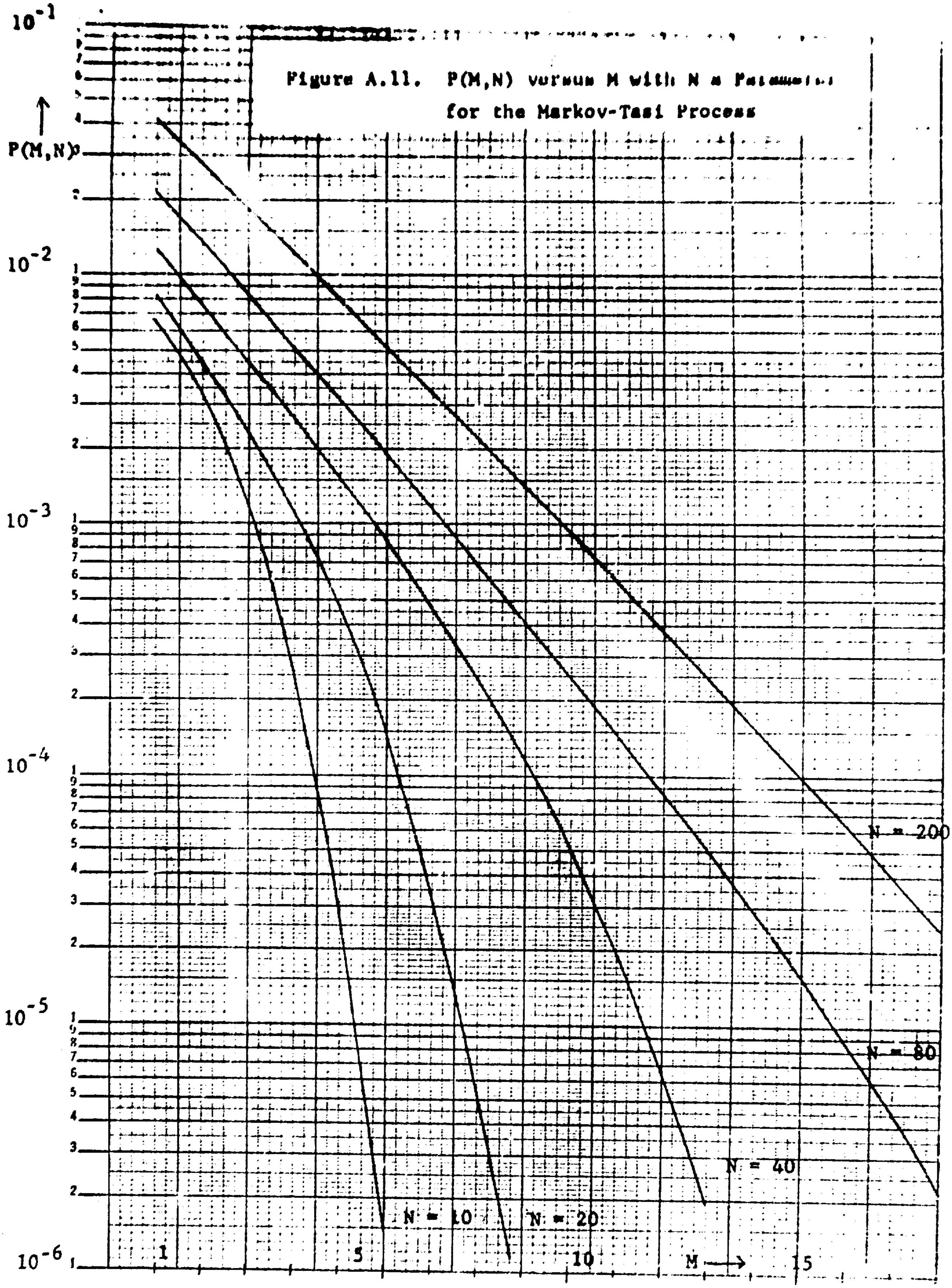


Figure A.11. P(M,N) versus M with N a Parameter for the Markov-Tasi Process

Figure A.12. S(M,N) versus N with M a Parameter for the Markov-Tsai Process $S(1,1) = 1;\ S(1,N) = 0 \quad N > 1$

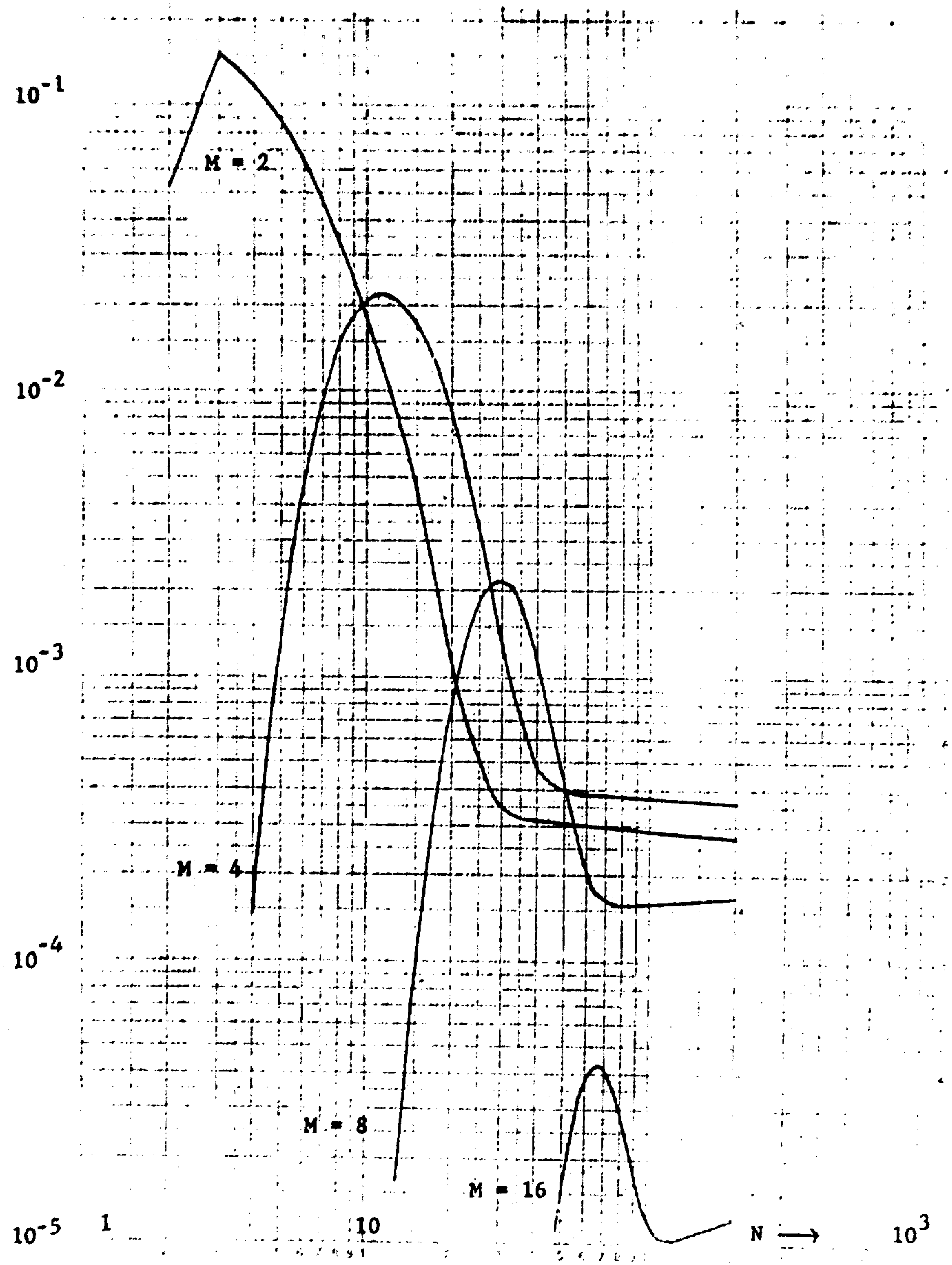


Figure A.13. S(M,N) Versus M with N a Parameter for the Markov-Tasi Process $S(1,1) = 1.0$; $S(1,N) = 0 \quad N > 1$

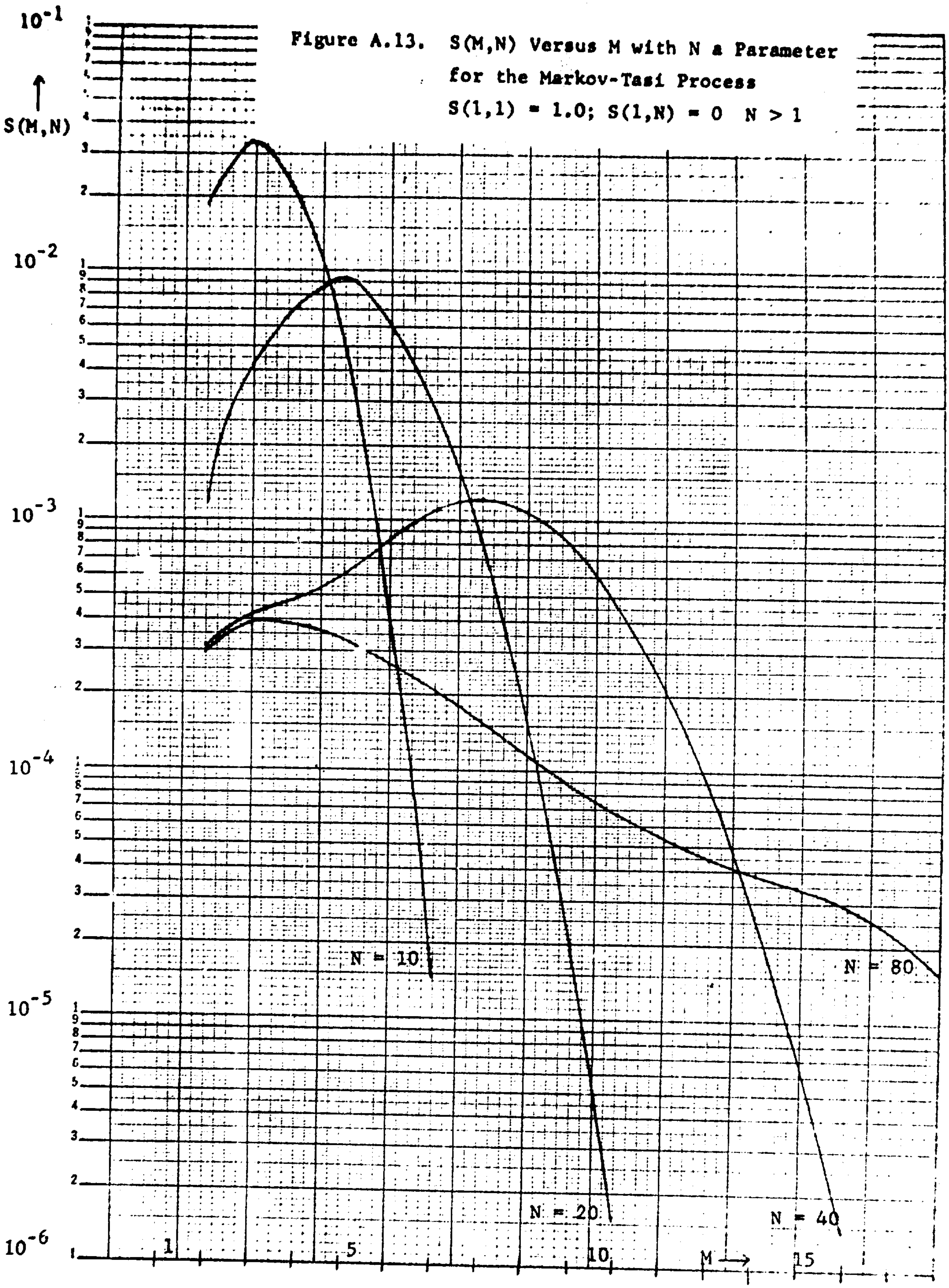


Figure A.14. S(M,N) versus M for the Binary Symmetric Channel

q = P(1) = 0.189; S(1,1) = 1.0; S(1,N) = 0, N > 1

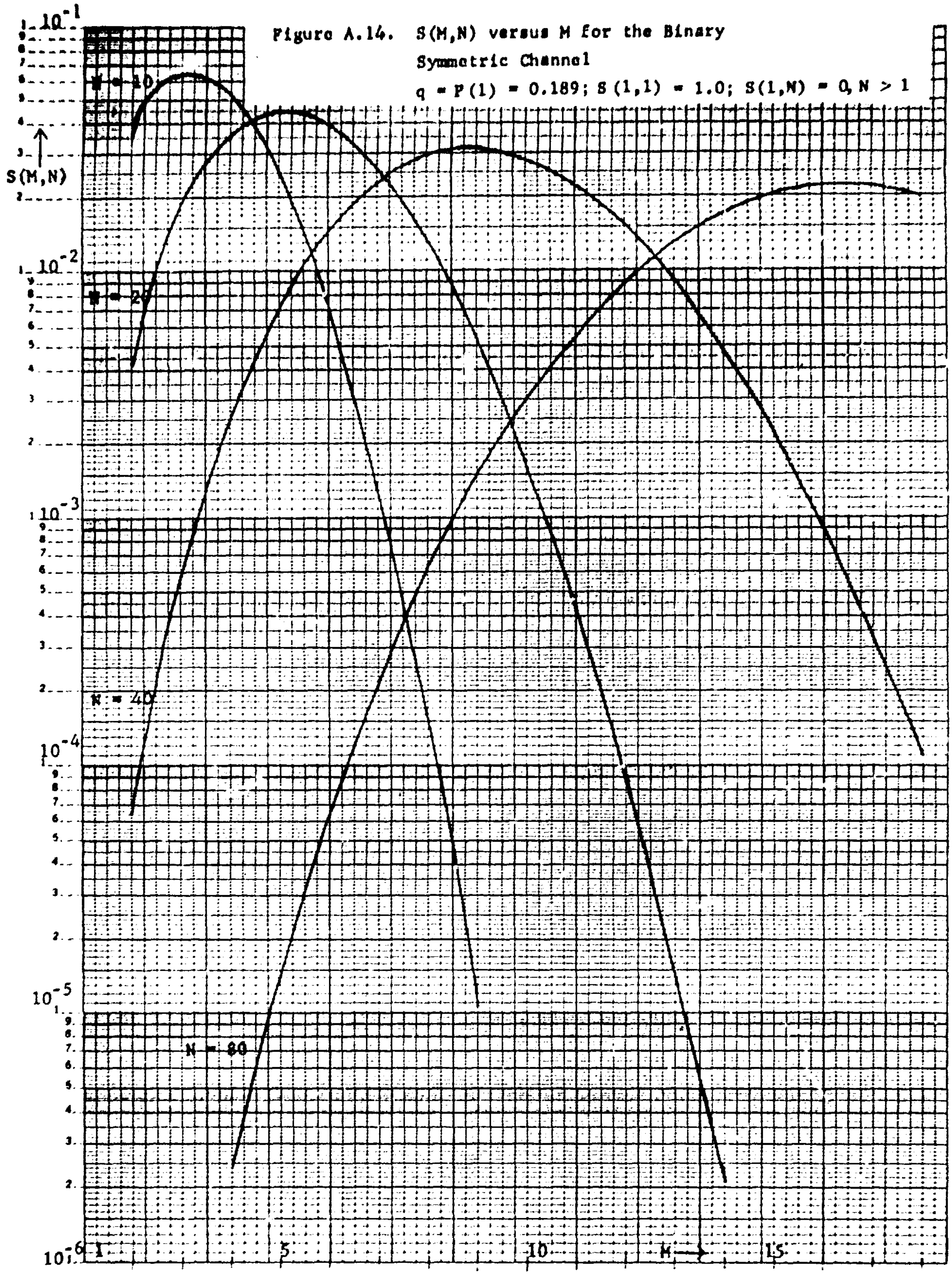


Figure A.15. S(M,N) versus N with M a Parameter for the Binary Symmetric Channel

$q = P(1) = 0.189;\ S(1,1) = 1.0;\ S(1,N) = 0\quad N > 1$

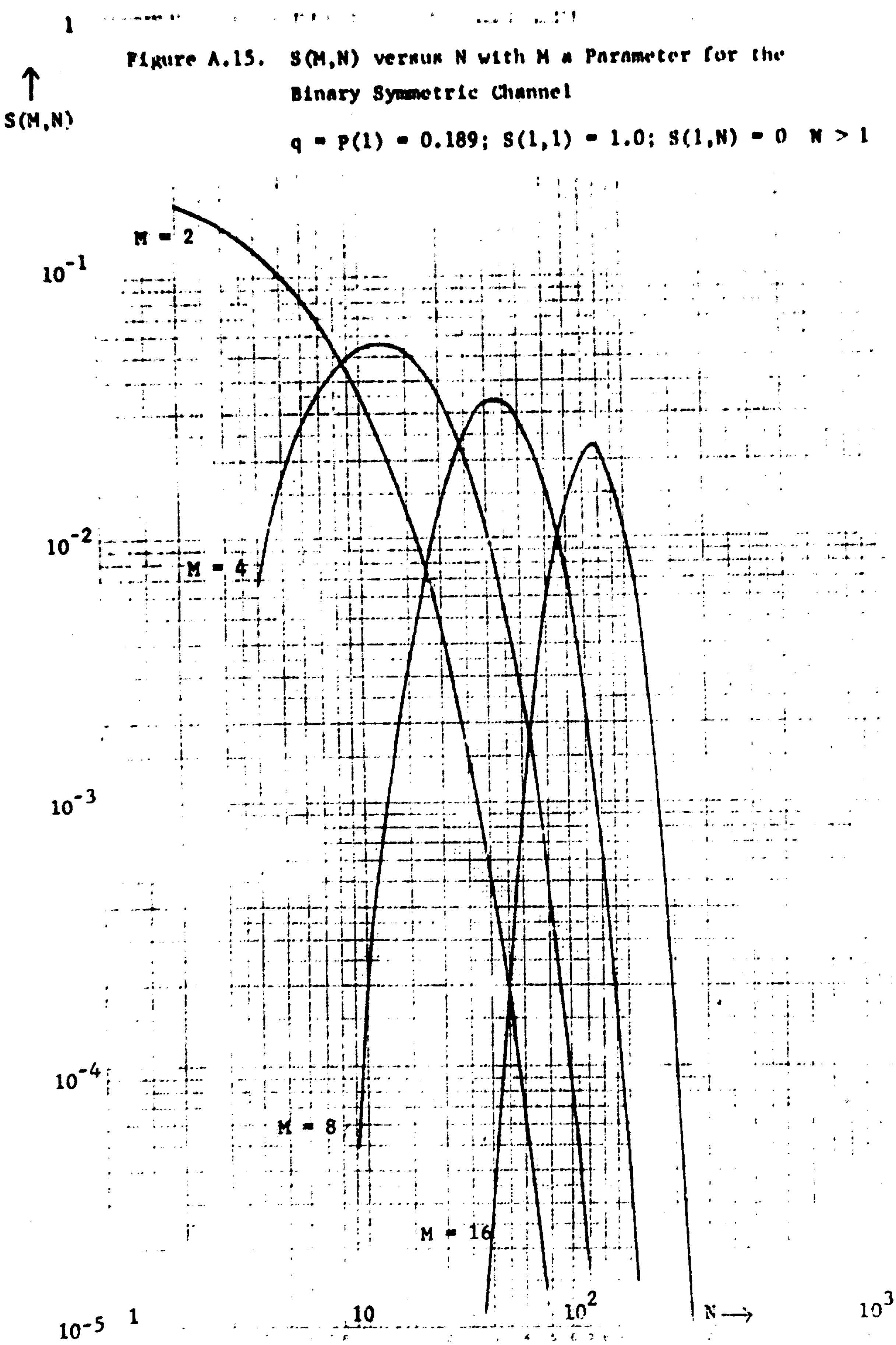


Figure A.16. a(b) and B(b,n)/n P(1) Versus b for n = 2000 for the Pareto Model.

B(b,n) - the probability of an error burst of length b in n bits

a(b) - the probability that the b*th* bit following an error is also an error

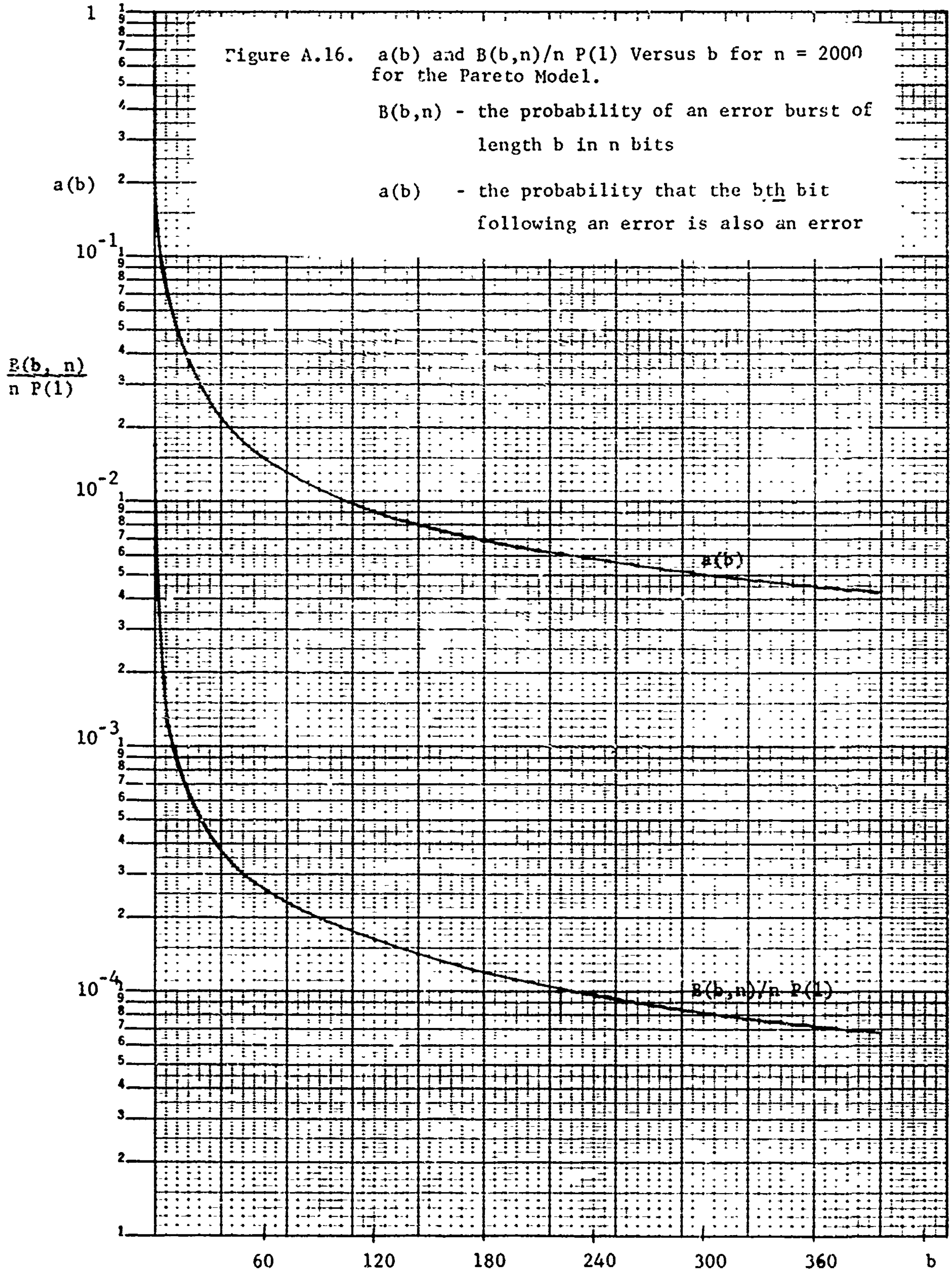
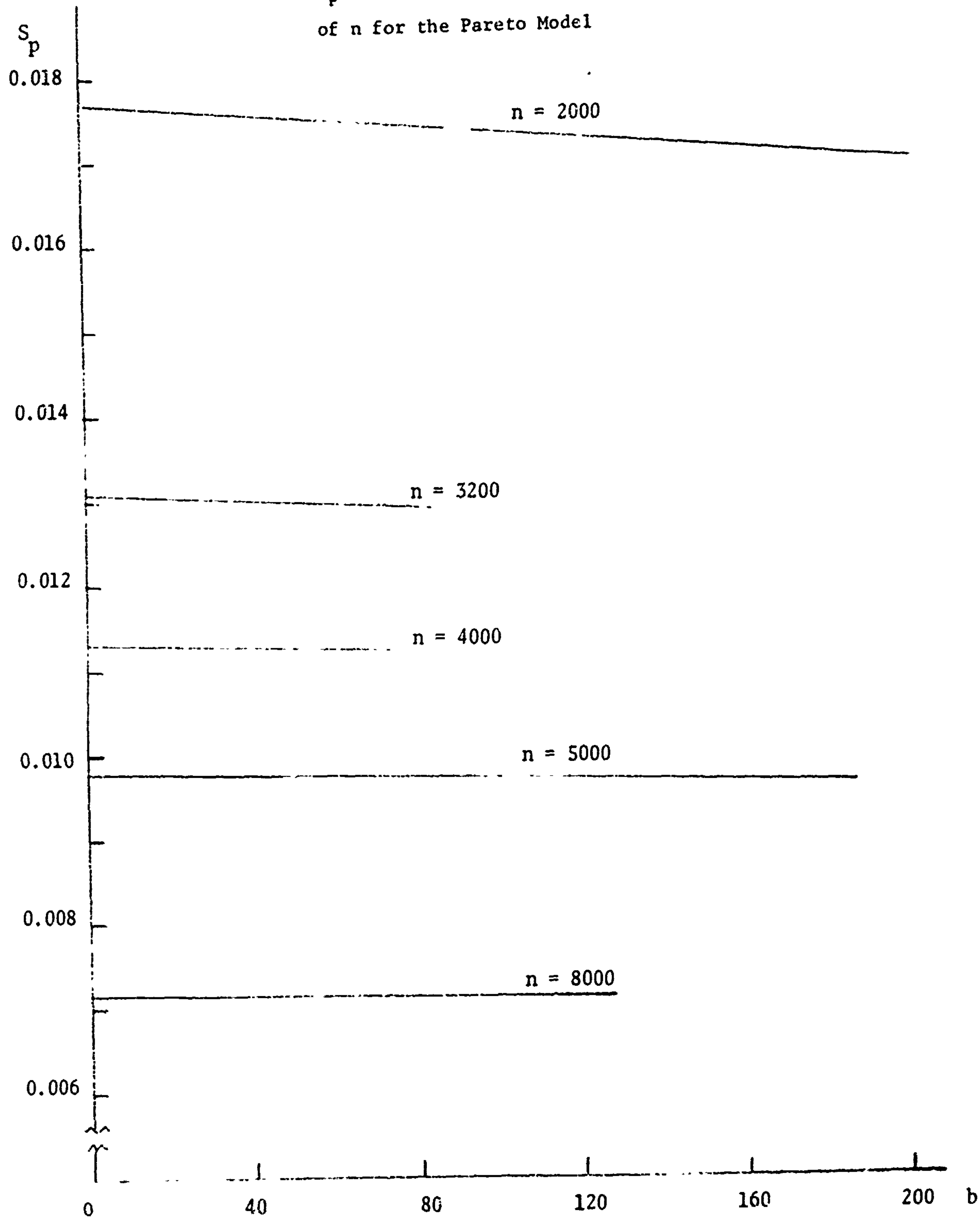


Figure A.17. $S_p(n,b)$ Versus b for Various Values of n for the Pareto Model



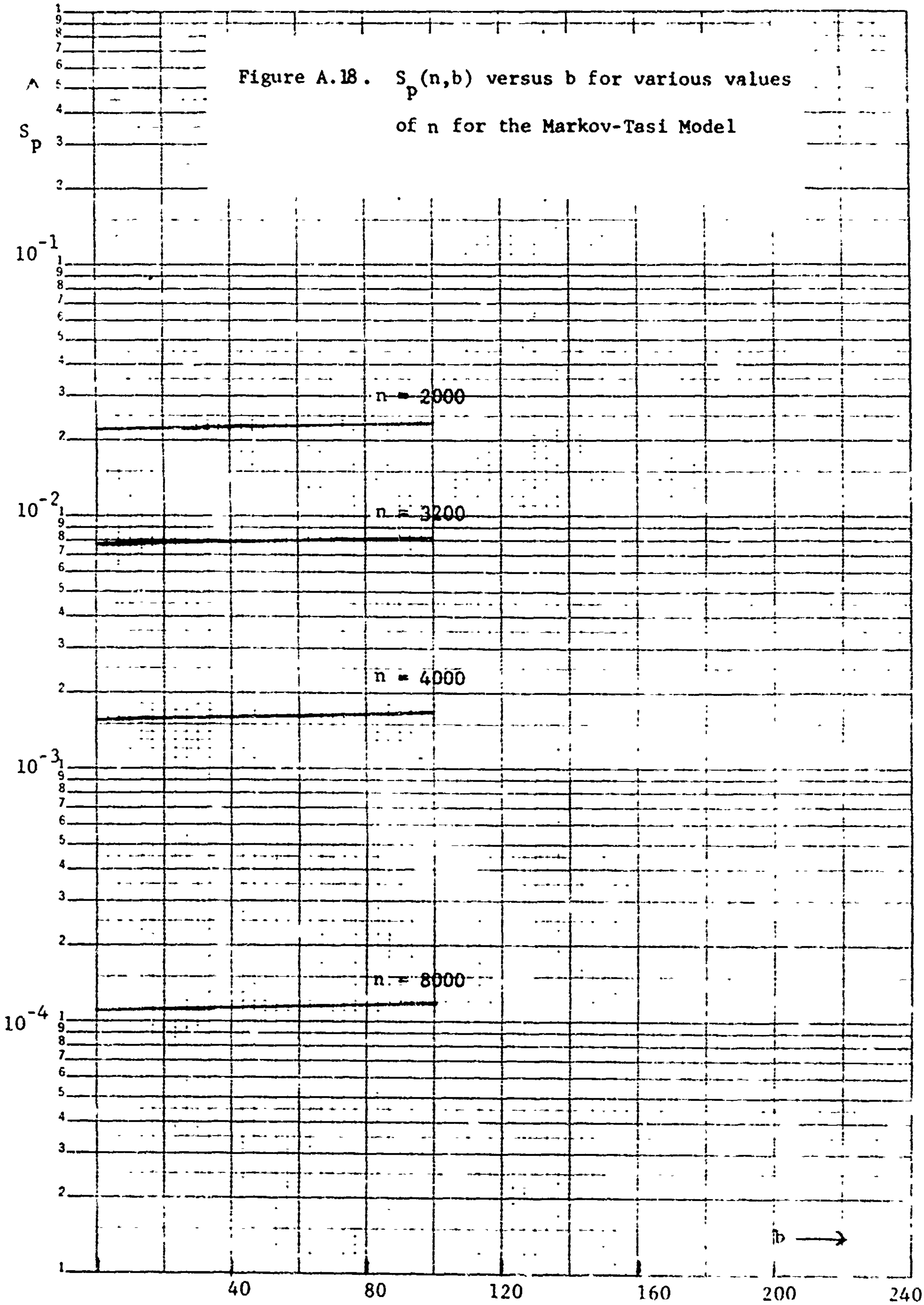


Figure A.18. $S_p(n,b)$ versus b for various values of n for the Markov-Tasi Model
$\hat{S}_p$
10^{-1}
10^{-2}
10^{-3}
10^{-4}
n = 2000
n = 3200
n = 4000
n = 8000
b →
40
80
120
160
200
240

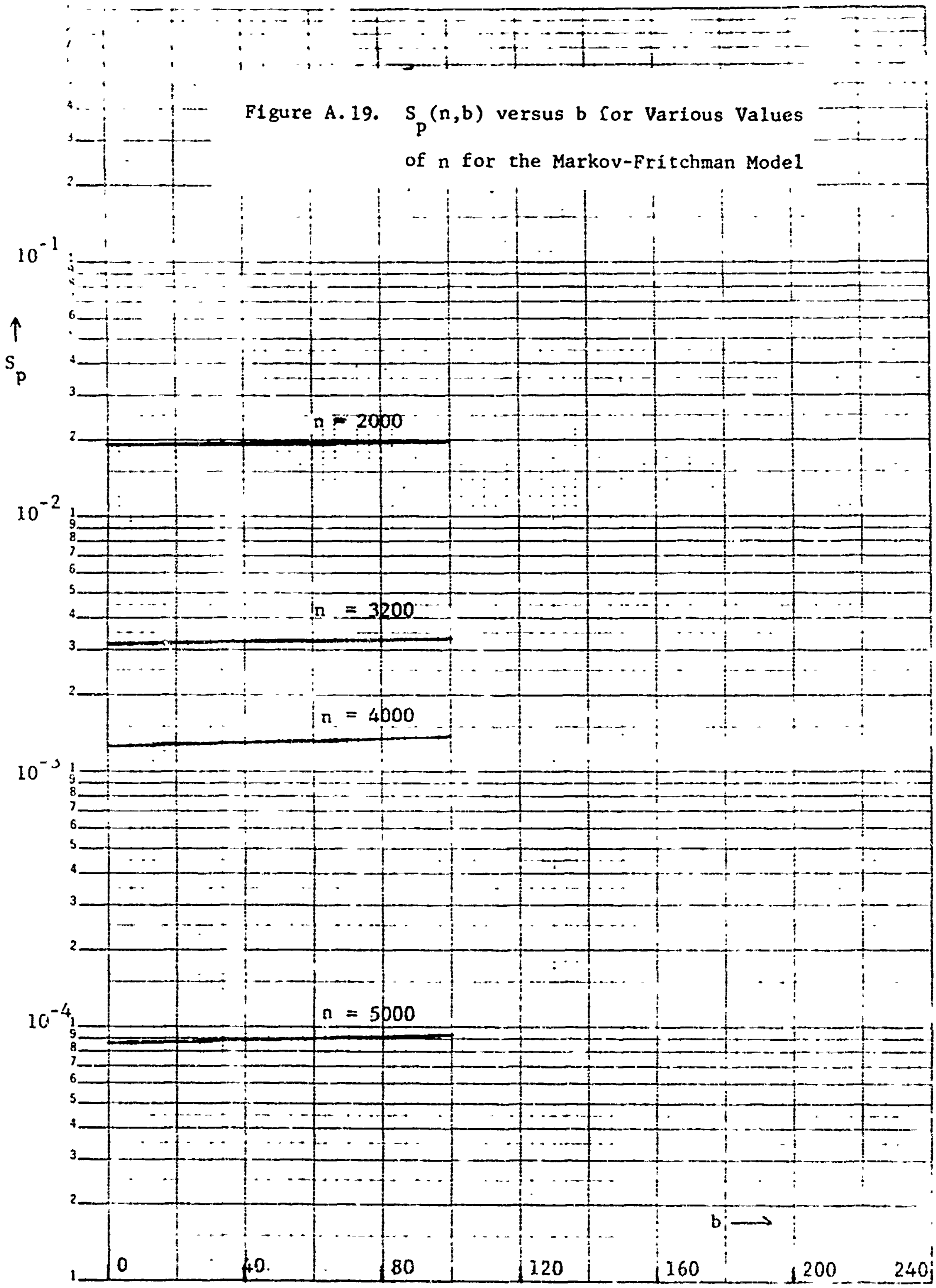


Figure A.19. $S_p(n,b)$ versus b for Various Values of n for the Markov-Fritchman Model

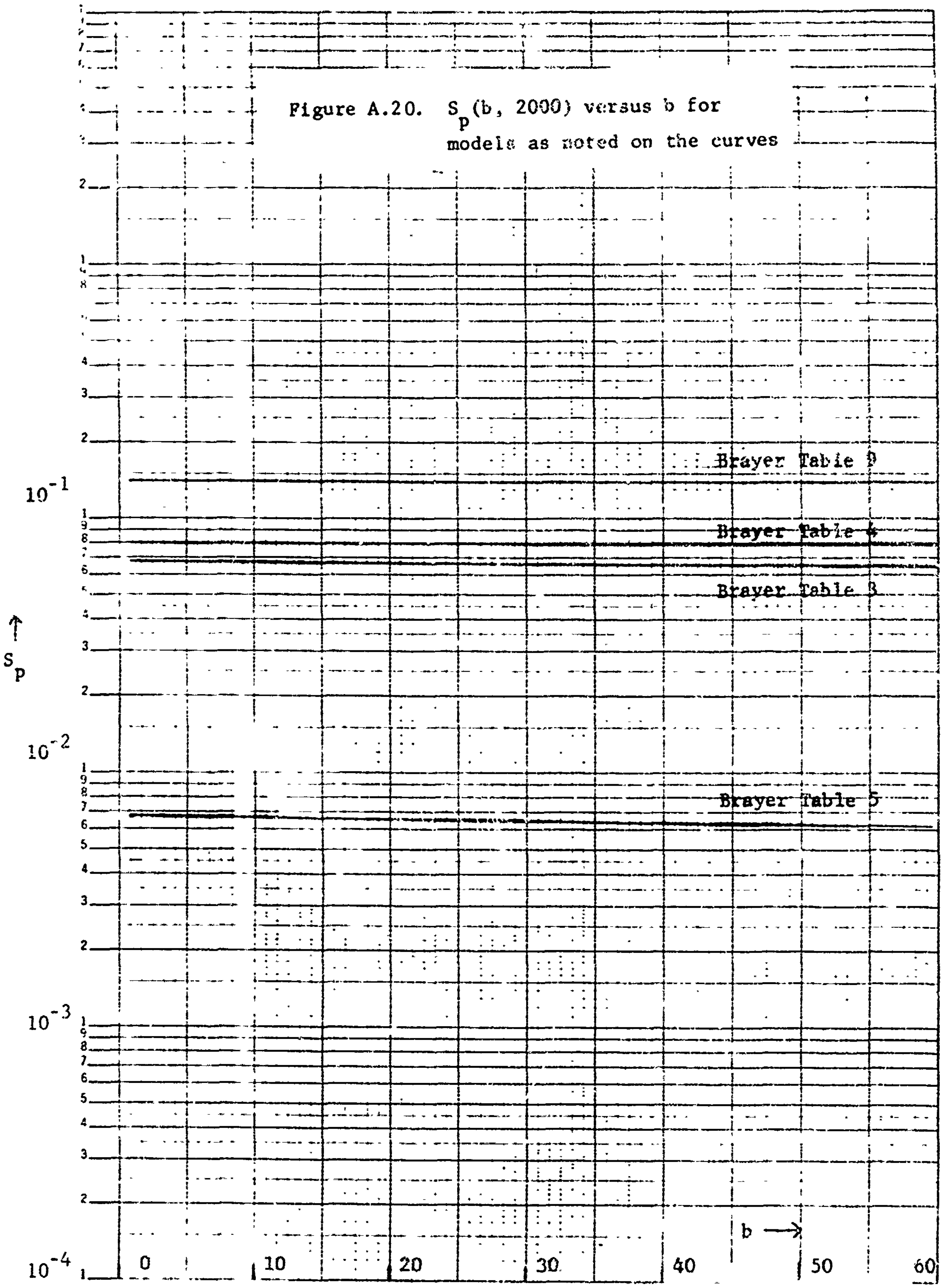


Figure A.20. S_p(b, 2000) versus b for models as noted on the curves
Brayer Table 9
Brayer Table 4
Brayer Table 3
Brayer Table 5
10^{-1}
10^{-2}
10^{-3}
10^{-4}
S_p
b
0
10
20
30
40
50
60

Figure A.21. P(M,N) versus N with M a Parameter for the Chien-Haddad Model A

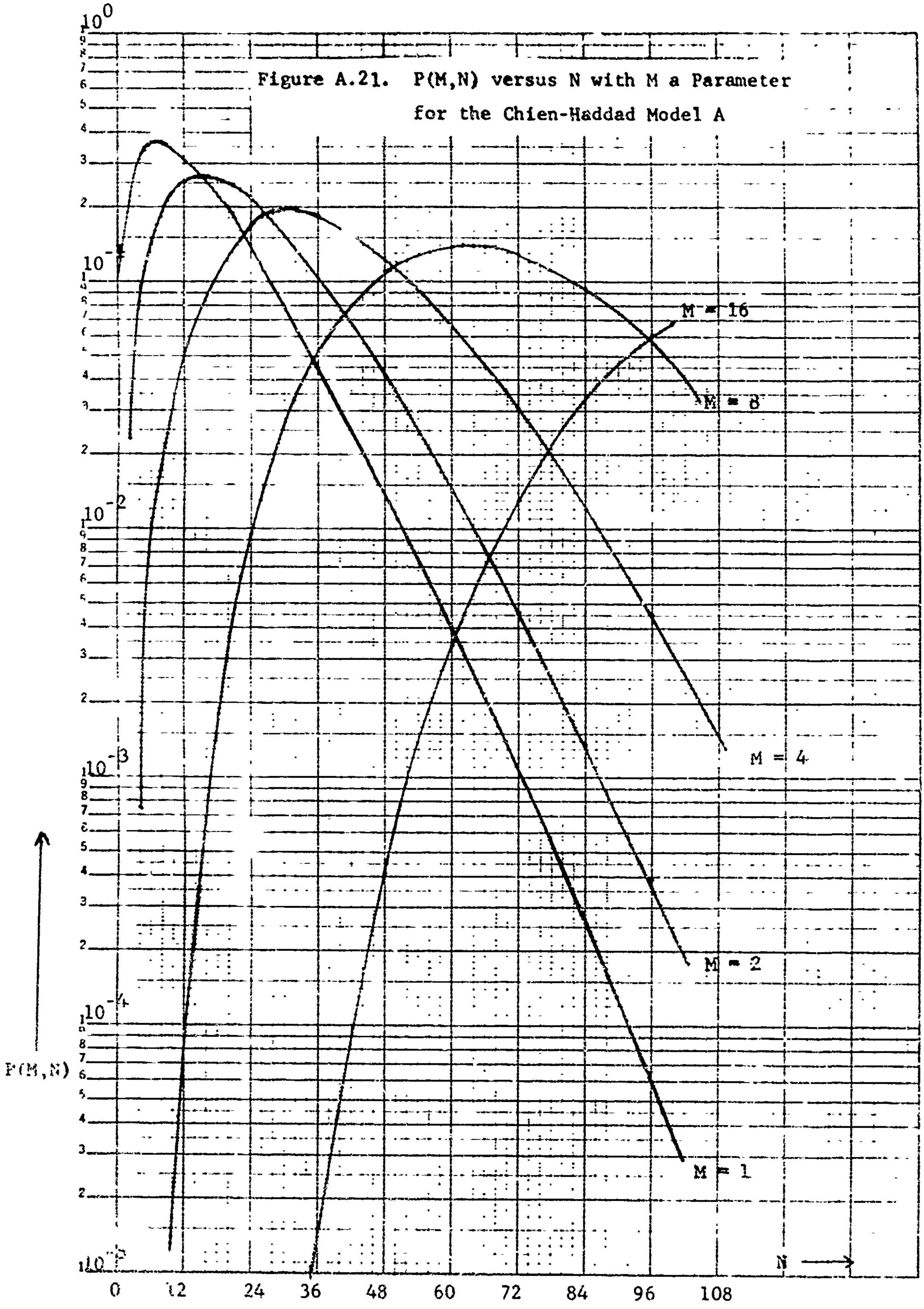
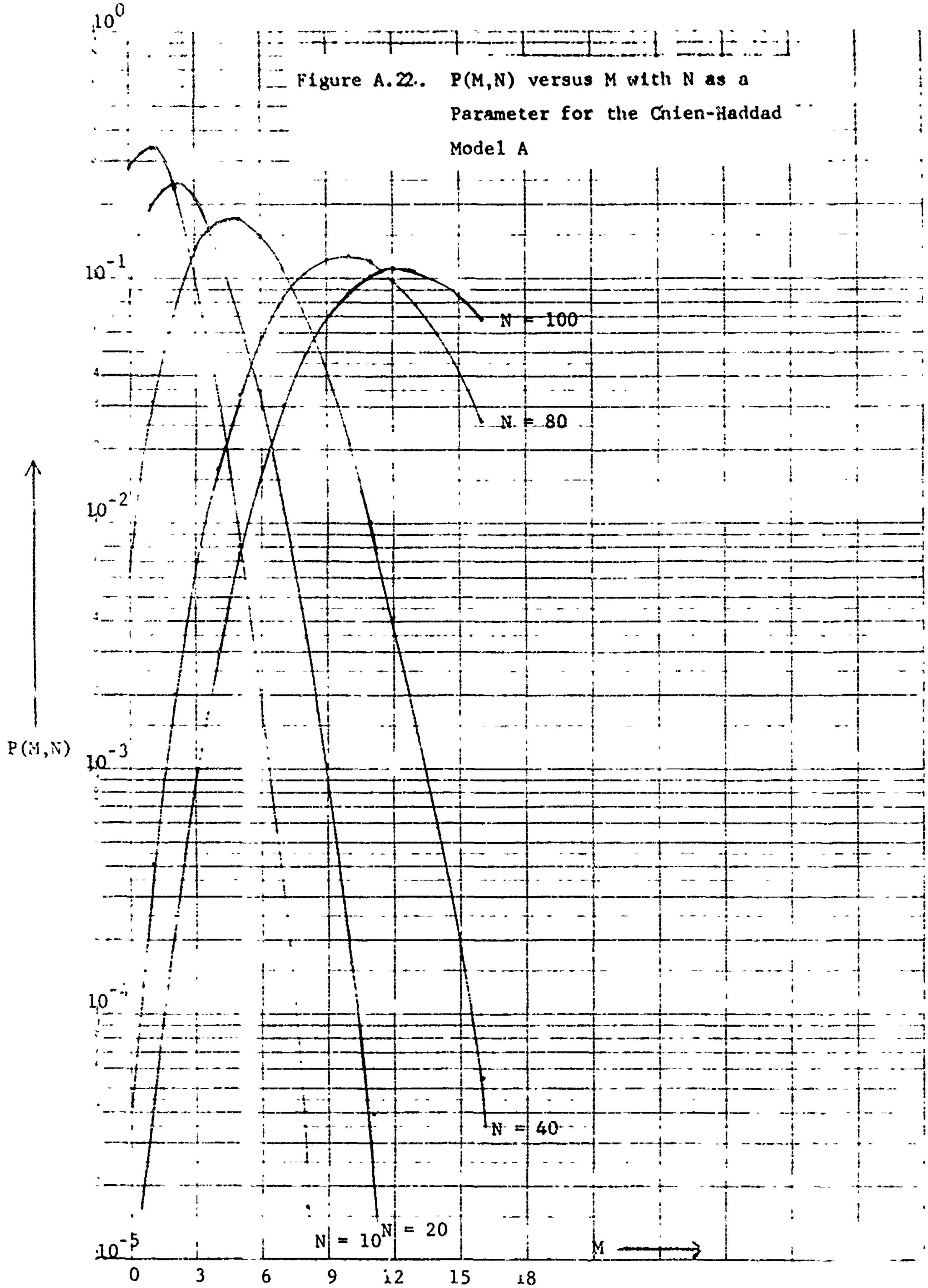


Figure A.22. P(M,N) versus M with N as a Parameter for the Chien-Haddad Model A



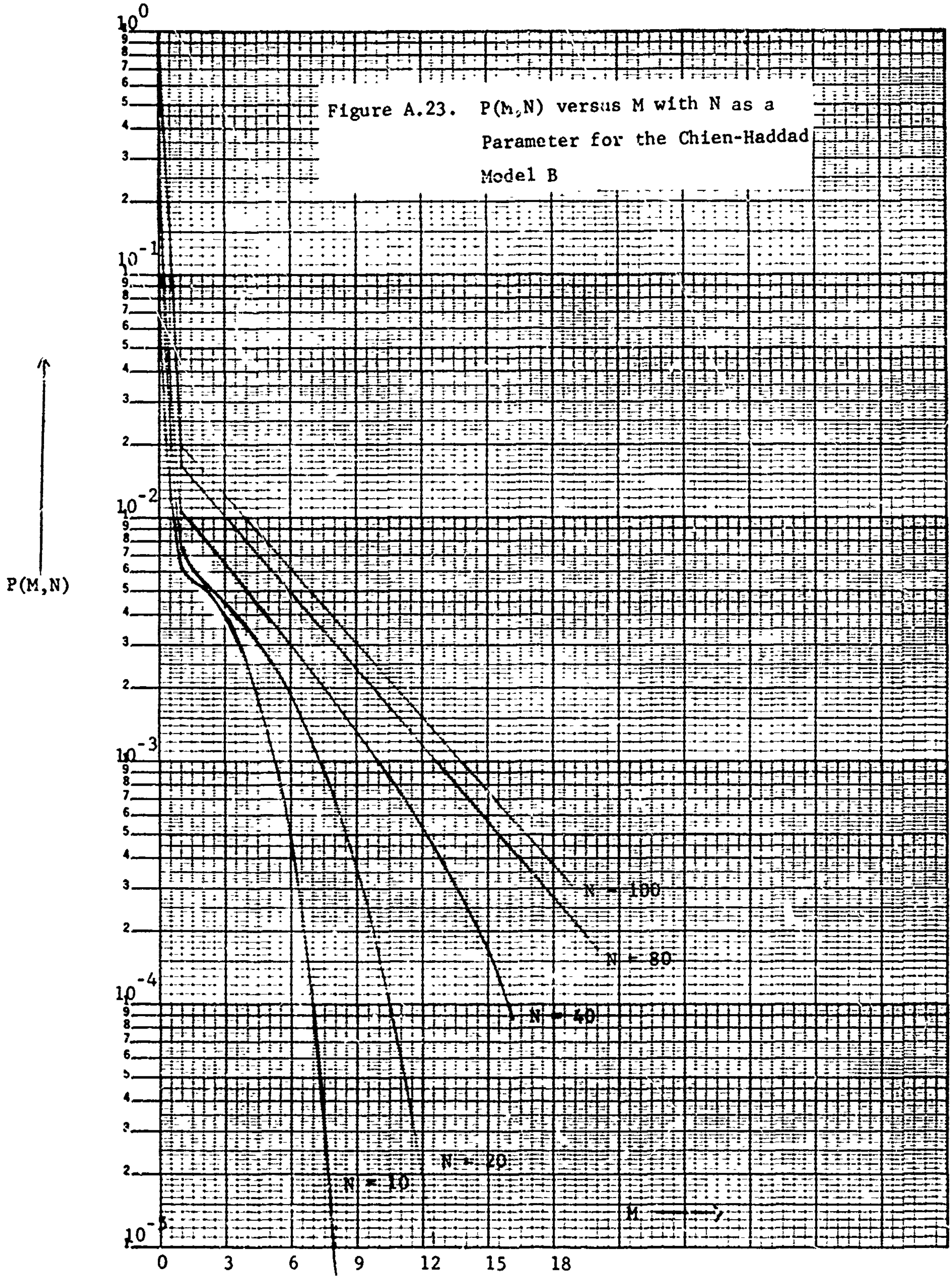


Figure A.23. P(M,N) versus M with N as a Parameter for the Chien-Haddad Model B

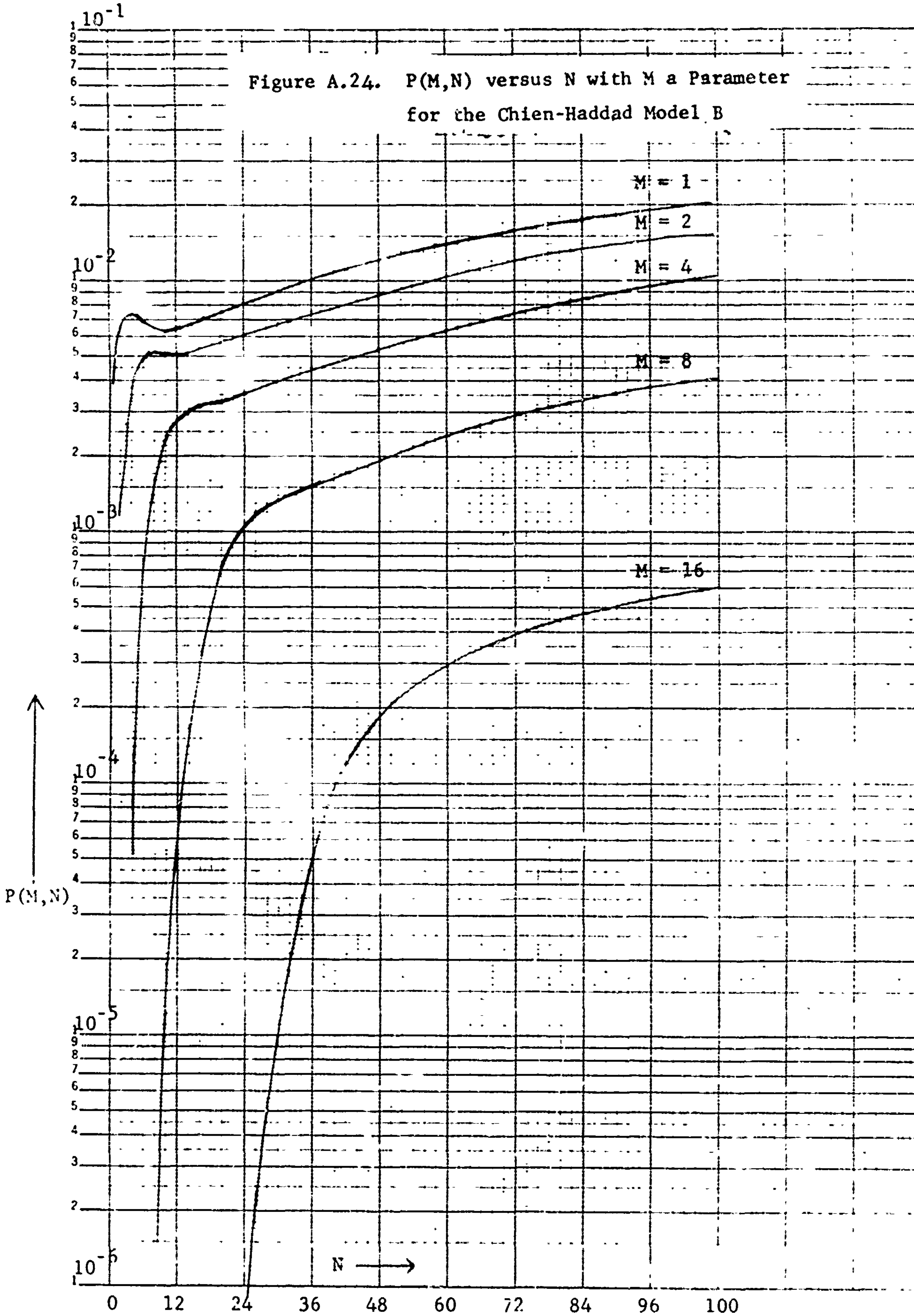


Figure A.24. P(M,N) versus N with M a Parameter for the Chien-Haddad Model B
M = 1
M = 2
M = 4
M = 8
M = 16
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{-5}
10^{-6}
P(M,N)
N
0
12
24
36
48
60
72
84
96
100

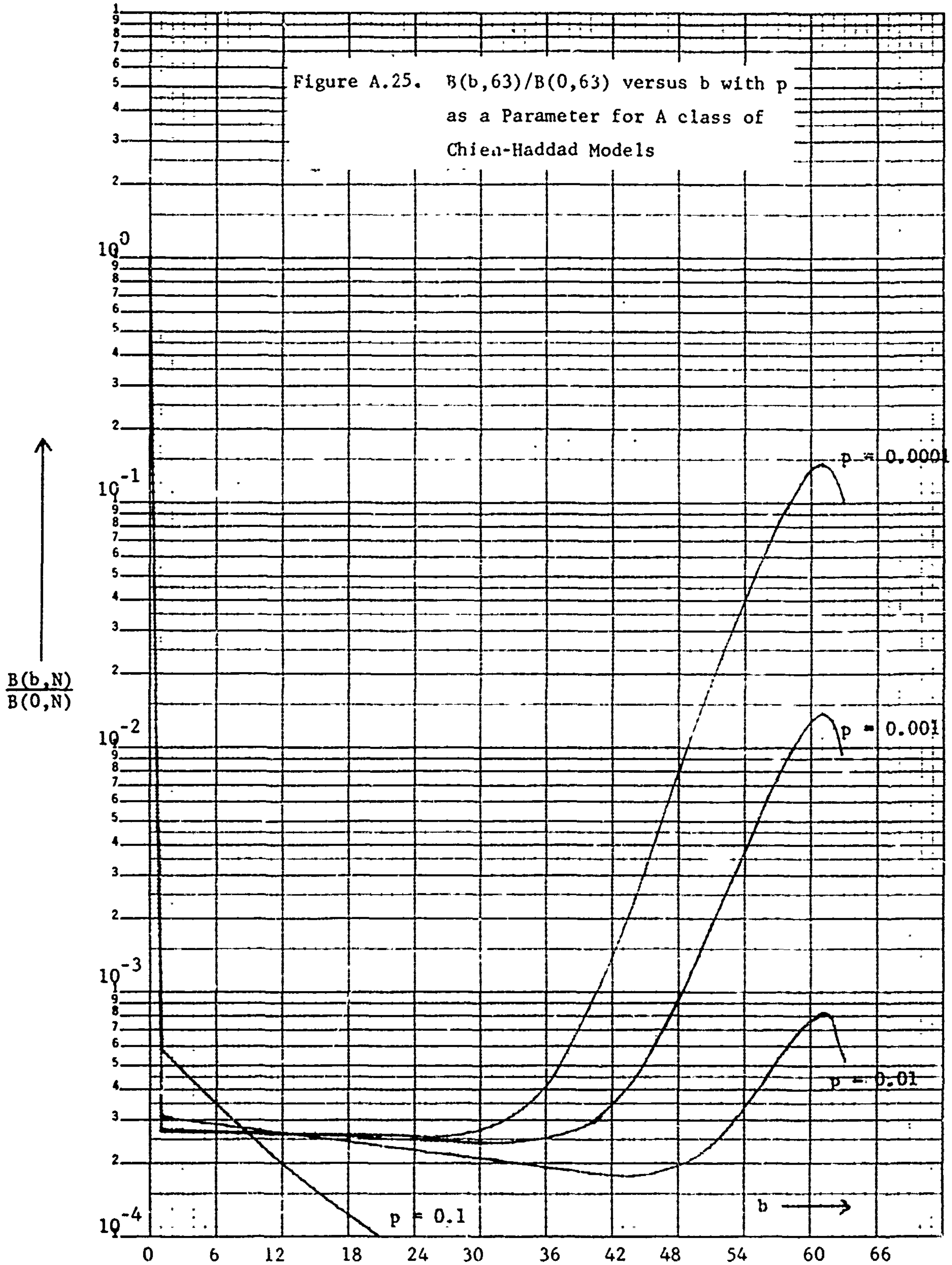


Figure A.25. B(b,63)/B(0,63) versus b with p as a Parameter for A class of Chien-Haddad Models

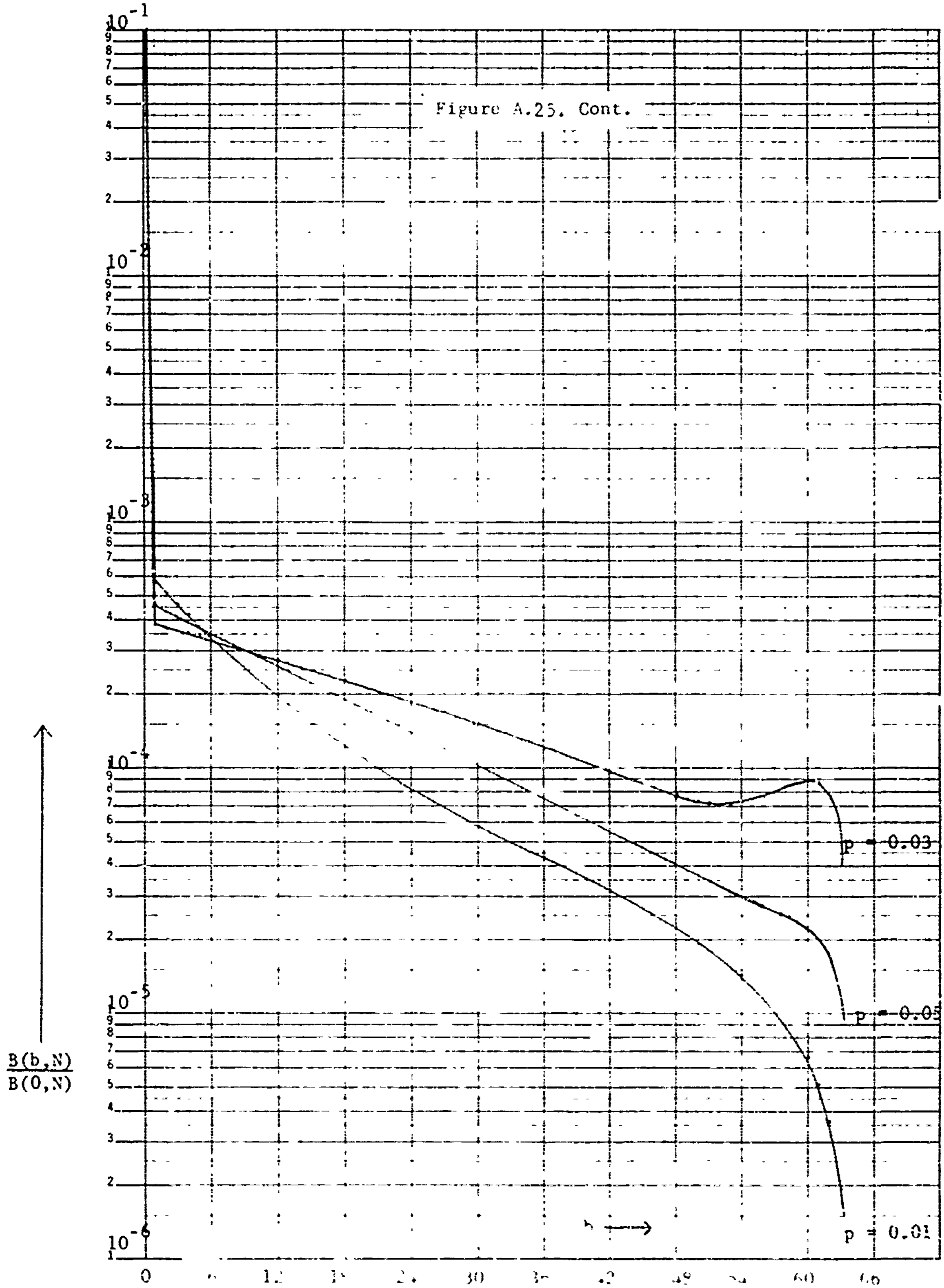


Figure A.25. Cont.
p = 0.03
p = 0.05
p = 0.01
b
B(b,N) / B(0,N)
0
12
30
60

APPENDIX II: PROGRAM MAINTENANCE MANUAL

SECTION 1. GENERAL DESCRIPTION

1.1 Purpose of PMM

The object for writing this PMM is to provide the maintenance programmer personnel with the information necessary to effectively maintain the system.

1.2 System Application

The system described in this manual consists of 11 independent program modules which are written for the evaluation of code generating polynomials.

Several error statistics are calculated for renewal channels to help evaluate the polynomials.

1.3 Equipment Environment

These 11 program modules were written for Univac 1108 Fortrain V Compiler and have been modified for IBM 360 Compiler H System.

To make use of the built-in logic functions, like LAND(a,b), LOR(a,b), LXOR(a,b), etc., an additional compiler option would be coded.

PARM. procstep = (....., XL, ...)

Here XL subparameter is not positional.

1.4 Conventions:

1) Integer Variables always begins with I, J, K, L, M, N

2) Abbreviation:

U.D. = Undetectable
ERR. = Error
PROG. = Program
Prob. = Probability
Info. Seq. = Information Sequence

SECTION 2. SYSTEM DESCRIPTION

2.1 General Description

Each of the 11 program modules is self-contained and can be compiled and linked to form independent load module.

Each program module contains at least one MAIN program. Some modules may contain one MAIN program and other subprograms.

The interaction between these program modules and the datasets is shown in Figure 1.

2.2 Detailed Description for Each Module

2.2.1 PROG. MODULE "Z"

a) Module Tag = Z

b) Given 1. weight of error burst
2. burst size
3. code generating polynomial

this module does exhaustive search for U.D. ERR. pattern, by doing polynomial division.

d) See comments on program list.

i) Subprograms ERPAT, DIVISN and FLD (J, K, MS, NV, NG, KP) are linked in this module.

For ERPAT and DIVISN, arguments are passed from MAIN program through COMMON block.

k) Stop execution, when I/O error occurs on card reader.

2.2.2 PROG. MODULE "A"

a) Module Tag = A

b) It creates and catalogs Dataset F(3080), P(3072) and R(128.200) for PARETO model with parameters ET = 3×10^4, $\gamma = 0.3$.

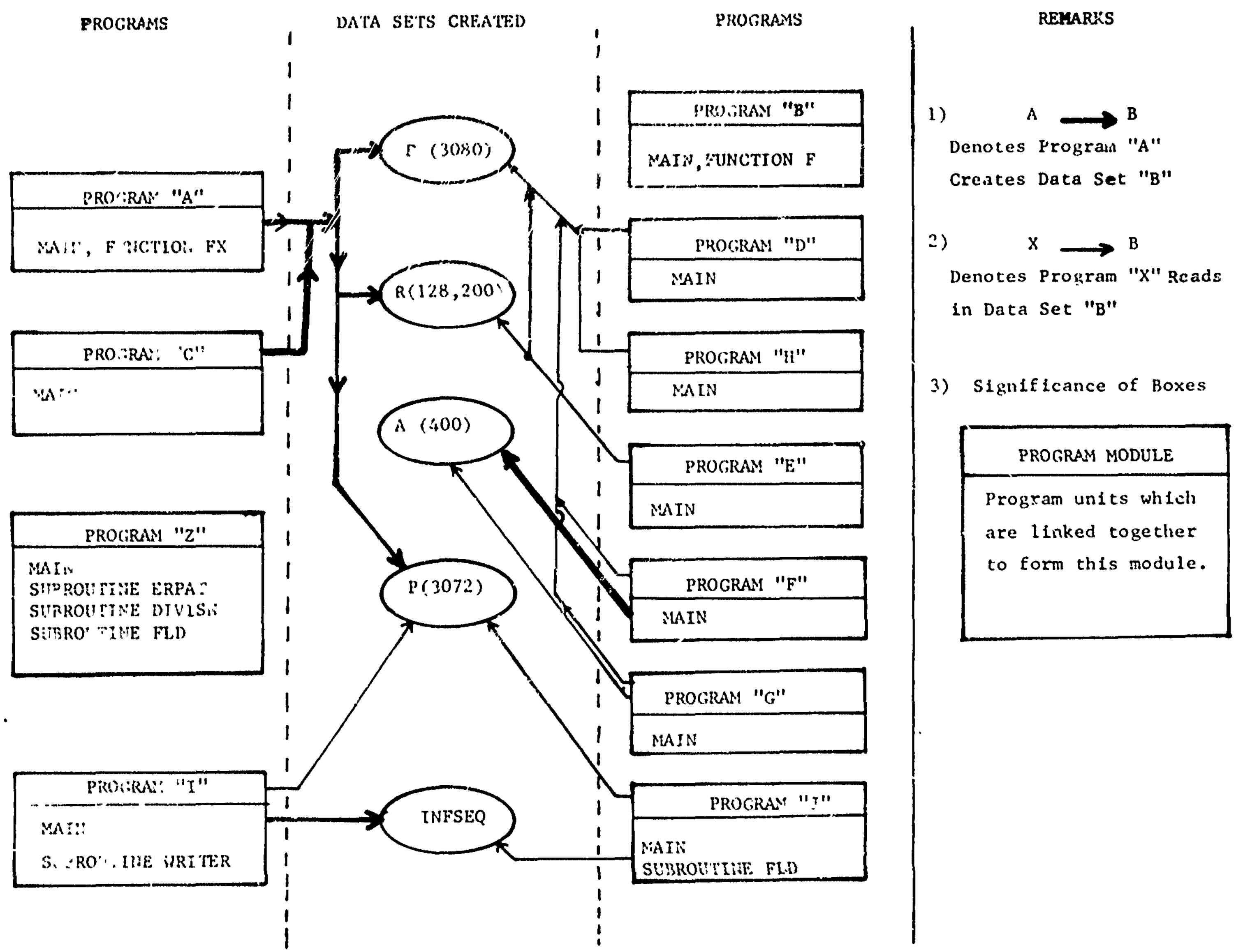


Figure 1. Structure of Computer Program

$$*F(n) = \text{Prob. } (0^{n-1}|1) = 1 - \left[\frac{1}{1-L^{-\alpha}}\right]\left[\frac{n^{\alpha}-1}{n^{\alpha}}\right], \quad n \geq 1$$

$$\text{here } L = \left[\frac{1-\alpha}{\alpha}(ET+1\right]^{\frac{1}{1-\alpha}}$$

$$*P(n) = \text{Prob. } (0^{n-1}\,1|1) = F(n) - F(n+1)$$

$$*R(m,n) = \begin{cases} F(n), & m = 1,\ n \geq 1 \\ \sum_{j=1}^{n-m+1} P(j)R(m-1,\ n-j), & 2 \leq m \leq n,\ n \geq 2 \end{cases}$$

*PE = individual error probability

$$= \frac{1-\alpha}{\alpha} L^{\alpha-1}$$

d) See commends on program list.

i) Subprogram Function FX(NA,SL,ALP) is linked with MAIN.

k) Stop execution when I/O error occurs on card reader.

2.2.3 PROG. MODULE "B"

a) Module Tage = B

b) This module calculates the U.D. ERR pattern's probability

$$\text{Prob.} = \prod_{i=2}^{W} [F(d_i) - F(d_i + 1)] \frac{1}{N} \sum_{d=1}^{N-b+1} F(d)F(N-b+2-d)$$

W is weight of burst

N is message block length

Refer to comments on program list

d) See comments on program list

h) Exit when card reader reaches the Delimieter Statement (/*).

k) Stop execution when I/O error occurs on card reader.

2.2.4 PROG. MODULE "C"

a) Module Tag = C

b) Creates and catalogs Datasets F, P and R for Markov models.

$$*F(n+1) = \text{Prob. } (0^n/1) = \sum_{K=1}^{N} t_{NK}(t_{KK})^{n-1}$$

$$*P(n) = \text{Prob. } (0^{n-1}1/1) = F(n) - F(n+1)$$

$$*R(m,n) = \begin{cases} F(n) & m = 1,\ n \geq 1 \\ \sum_{j=1}^{n-m+1} P(j)R(m-1,n-j) & 2 \leq m \leq n \end{cases}$$

$$*PE(\text{individual error probability}) = \left[1 + \sum_{K=1}^{N-1} \frac{t_{NK}}{1-t_{KK}}\right]^{-1}$$

N - number of states

t_{NK} - entry at Nth row and Kth column of state-transition matrix

d) See comments on program list.

k) Stop execution when I/O error occures on card reader.

2.2.5 PROG. MODULE "D"

a) Module Tag = D

b) Calculates the U.D. ERR. pattern probability for Markov models.

$$\text{Prob.} = \prod_{i=2}^{W} [F(di) - F(di+1)] \frac{1}{N} \sum_{d=1}^{N-b+1} F(d)\ F(N-b+2-d)$$

W = weight of error burst.

di's = gap length of the pattern

N = message block length

Refer to comments on program list.

d) See comments on program list.

h) Exit when card reader reaches the Delimiter Statement (/*).

k) Stop execution when I/O error occures on card reader.

2.2.6 PROG. MODULE "E"

a) Module Tag = E

b) Calculates P(m,n) for both PARETO and MARKOV models.

$$P(m,n) = \sum_{j=1}^{n-m+1} PE \cdot F(j) \cdot R(m,n-j+1) \quad 1 \le m \le n$$

PE(individual error prob.), F(j) and R(x,y) are all created in module A or C.

d) See comments on program list.

h) Exit when card reader reaches the Delimiter Statement (/*).

2.2.7 PROG. MODULE "F"

a) Module Tag = F

b) It creates and catalogs Dataset A(j) for the use of module G. Applicable to both Markov and Pareto models.

$$A(j) = \begin{cases} 1 & j=0 \\ F(1) - F(2) & j=1 \\ [F(j) - F(j+1)] + \sum_{s=1}^{j-1} [F(s) - F(s+1)] A(j-2) & j \ge 1 \end{cases}$$

d) See comments on program list.

2.2.8 PROG. MODULE "G"

a) Module Tag = G

b) Calculate (B(b,N)/N•PE) for Markov and Pareto models.

$$\frac{B(b,N)}{N \cdot PE} = A(b-1) \cdot \frac{1}{N} \sum_{d=1}^{N-b+1} F(d)F(N-b+2-d)$$

A(x) is autocorrelation array created in module F.

d) See comments on program list.

2.2.9 PROG. MODULE "H"

a) Module Tag = H

b) Calculate quantity Sp(b,N)

$$Sp(b,N) = \frac{1}{N} \sum_{d=1}^{N-b+1} F(d)\ F(N-b+2-d)$$

d) See comments on program list.

Only 1 input data card, it contains KB (limit of b) and N(block length).

This module will print Sp(1,N) to Sp(KB,N).

2.2.10 PROG. MODULE "I"

a) Module Tag = I

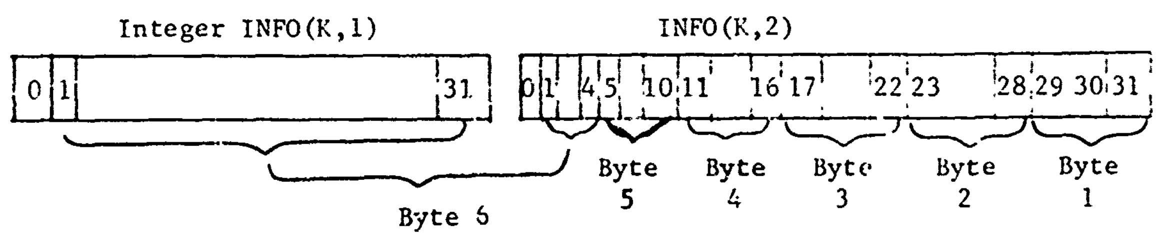
b) This module generates most probable information sequence based on Pareto model's gap statistics P(3072).

The info. seq. is used in module J to evaluate code generating polynomials.

The info. seq. generated is stored in a 2 dimensional array INFO(1000,2) before its being written to Dataset INFSEQ.

The Kth info. seq. is stored as follows:

Assume Kth info. seq. is 1 0 0 0110 0 0100 10 (weight = 5)



Note: Bit 0 is not used to construct byte 6.

Then Byte 1 = 5 = weight

Byte 2 = 2 = position of right most "1" in this info. seq.

Byte 3 = 5 = position of 2nd right "1" in this info. seq.

Byte 4 = 9 = position of 3rd right "1" in this info. seq.

Byte 5 = 10 = position of 4th right "1" in this info. seq.

Byte 6 = 14 = position of left most "1" in this info. seq.

Note: For weight other than 5, the byte allocations are different from above.

d) See comments on program list.

i) Subroutine WRITER (ICT,II) is link ' with MAIN program in this module.

2.2.11 PROG. MODULE "J"

a) Module Tag = J

b) This module evaluates polynomials according to the following steps:

1) Read in info. seq. (created in module I) and P(3072) dataset (created in Module A or C)

2) Read in a polynomial G(x)

If a Delimiter Statement (/*) is read, go to Step 6.

3) For each info. seq., get a U.D. ERR. pattern which is $INF \cdot X^K + R(x)$.

Here INF is info. seq.

K is degree of G(x)

R(x) is the remainder of $(INF \cdot X^K/G(x))$.

4) Calculate probabilities of U.D. ERR. patterns obtained in Step 3 and sum it up for all info. seq.

5) Jump back to Step 2 to read one more polynomial.

6) Arrange the polynomials in ascending order according to the total U.D. ERR. probability associated with it.

7) Print the polynomials and its probability in ascending order.

d) See comments on program list.

h) Exit when card reader reaches /* statement.

i) Subroutine FLD(J, K, MS, NV, NG, KP) is linked with MAIN program.

k) Stop execution when I/O error occures on card reader.

SECTION 3. INPUT/OUTPUT DESCRIPTIONS

3.1 General Description

This system uses 5 I/O data sets -- F, P, R, A, and INFSEQ.

These datasets can be created on Tape or other secondary storage.

The reference number used for each data set is indicated in the comments of each program.

*Datasets F, P. R and A are created under Format

(5X, 5(E23.16,2X))

*Dataset INFSEQ is created without format control.

SECTION 4. PROGRAM ASSEMBLING, LOADING

a) To obtain load modules for each program-module described in Section 2, please refer to

"IBM SYSTEM 360, FORTRAN (G&H) PROGRAMER'S GUIDE"
GC28-6817-3 Page 83 .

b) To specify a dataset for a run, please refer to the same document as above pages 49 - 52.

c) The test runs for modules Z, B, F, G, H, I and J are described below

Remark:

*The polynomials shown on the Univac 1108's output are in Octal representation. For IBM 360 polynomials will be in Hexadecimal representation.

*b = blank

Assume Datasets F. P. R. A for Pareto model and Brayer's table 6 model are already created.

1) Module Z.

1st input card = bb6b15

2nd input card = 000E04B
= 160113 (Octal)

3rd input card = b16b29

The output is shown on page A-1

2) Module B

1st input card = b3200

2nd card = bbbb4bbbb4bbb....

3rd card = bbb18bbb47bbb61bb...

4th card = bbb39bbb65bbb70bb...

5th card = bbb68bbb71bbb90bb...

6th card = bbb77bbb89bbb93bb...

7th card = /*

Output is shown on page A-2

3) Module F

Specify Pareto model's Dataset F as input dataset with reference number = 10.

No input datacard.

Part of the output is shown on page A-3.

4) Module G

Specify Pareto Model's Dataset F and A as input datasets with reference numbers = 10, 11 respectively

1st input data card = 2000bbb...

Part of the output is shown on page A-4

5) Module H

Specify Pareto Model's Dataset F as input dataset with reference number = 10.

1st input data card = bb452000bb..

The output is shown on page A-5

6) Module I

Specify Pareto model's Dataset P as input dataset with reference number = 9

The output is shown on page A-6.

7) Module J

Specify Brayer Table 6 model's Dataset P and INFSEQ (created in Module I test run) as input dataset with reference numbers = 9,8 respectively.

1st card = 32bbb...

2nd card = bbbbbbb104C11Db7bbb...

[= 40460216667 (Octal)]

3rd card = bbbbbbb19262E7C59bbb...

[= 62613476131 (Octal)]

4th card = bbbbbbb1857D984Dbbb...

[= 60537314115 (Octal)]

5th card = /*

The output is shown on page A-7.

For polynomial = $(0000E04B)_{16}$ = $(160113)_8$; Number of error bits = 6

```
burst No. of      THE FOLLOWING ERROR PATTERNS ARE UNDETECTABLE:
size  error bits  EXPONENTS OF NON-ZERO TERMS
 ↓     ↓
( 16,  6)
( 17,  6)
( 18,  6)
( 19,  6)
( 20,  6)
( 21,  6)
( 22,  6)
( 23,  6)
             0     9    12    15    18    22
( 24,  6)
( 25,  6)
( 26,  6)
             0     3     9    21    22    25
             0    11    13    16    18    25
( 27,  6)
( 28,  6)
( 29,  6)
ESTIMATED RUN TIME EXCEEDED
 REENT ADDR:012453  BDI:000004

     X                        000000 000007  000000 000000  000000 000000  000000 0
          000000 000000  000000 000000  000000 000000  000164 012176  777777 7

     A    777777 777771  000000 134425  000000 071052  000000 000071  777777 7
          000000 000000  000000 000000  000000 000000  000000 000000  000000 0
          000000 000000  000000 000000

     R                        000000 000002  777777 777776  000164 044516  000000 0
          000000 000000  000000 000000  000000 000000  777777 777776  000000 0
RUNSTREAM ANALYSIS TERMINATED


  RUNID: EEASD     ACCT: 01,A0134        PROJECT: MURTHY-V-R

      EEASD   MAX TIME

  TIME:      TOTAL:  00:01:00.723

             CPU:    00:00:58.376       I/O:   00:00:01.395

             CC/ER:  00:00:00.951       WAIT:  00:00:00.000

  IMAGES   READ:    7           PAGES:   2

  START:     22:34:09 JUL 29,1974       FIN:   22:35:25 JUL 29,1974
```

```
*******160113 POLYNOMIAL 3-BIT,4-BIT,5-BIT,6-BIT ******
            N = 3200
BAGT GELGEE.FANTOM
   0   18   47   61
  PROBABILITY=   .000000003054
   0   39   65   70
  PROBABILITY=   .000000004674
   0   68   71   96          These 4 patterns are tested
  PROBABILITY=   .000000006150
   0   77   89   93
  PROBABILITY=   .000000006709
   0   10   18   23   24
  PROBABILITY=   .000000021692
   0   13   20   26   34
  PROBABILITY=   .000000001473
   0    6   15   26   35
  PROBABILITY=   .000000001158
   0   10   28   39   47
  PROBABILITY=   .000000000300
   0   20   36   46   48
  PROBABILITY=   .000000000622
   0   22   32   44   49
  PROBABILITY=   .000000000369
   0    9   30   42   49
  PROBABILITY=   .000000000297
   0   13   14   37   51
  PROBABILITY=   .000000001184
   0    8   18   29   51
  PROBABILITY=   .000000000233
   0   21   39   43   53
  PROBABILITY=   .000000000305
   0   18   27   40   53
  PROBABILITY=   .000000000152
   0   30   33   36   53
  PROBABILITY=   .000000001221
   0    4   35   53   55
  PROBABILITY=   .000000001224
   0   18   37   55   60
  PROBABILITY=   .000000000127
   0    3   36   55   61
  PROBABILITY=   .000000000161
   0   12   27   38   61
  PROBABILITY=   .000000000039
   0   30   45   57   63
  PROBABILITY=   .000000000120
   0    3   24   41   63
  PROBABILITY=   .000000000169
   0   14   26   45   64
  PROBABILITY=   .000000000056
   0    8   11   54   66
  PROBABILITY=   .000000000372
   0   10   15   53   66
  PROBABILITY=   .000000000156
   0    9   12   15   18   22
  PROBABILITY=   .000000001970
   0    3    9   21   22   25
  PROBABILITY=   .000000002533
   0   11   13   16   18   25
  PROBABILITY=   .000000001927
```

**** A(B-1) PARETO DISTRIBUTION --ET=30000

[illegible]/FOR64	1	001063	001210	2	040012	040054
[illegible]/FOR	1	001211	001233			
[illegible]/FOR68	1	001234	001455	2	040055	040151
[illegible]/FOR				2	040152	042353
[illegible]/FOR69-E	1	001456	001723	2	042354	042410
[illegible]/FOR69-A	1	001724	002003	2	042411	042424
[illegible]/FOR69	1	002004	002030			
[illegible]/FOR69	1	002031	002142			
[illegible]/FOR69	1	002143	002203			
[illegible]/FOR69	1	002204	002237			
[illegible]/FOR68	1	002240	002262			
[illegible]/FOR68	1	002263	002557	2	042425	042430
[illegible]/FOR69	1	002560	003734	2	042431	042467
[illegible]/FOR69	1	003735	004611	2	042470	042544
[illegible]/FOR69-D	1	004612	005014	2	042545	042703
[illegible]/FOR69-D	1	005015	006002	2	042704	043057
				4	043060	043131
[illegible]/FOR				2	043132	043170
ERU$/SYS69-S						
[illegible]/FOR69-A	1	006003	006211	2	043171	043210
[illegible]/FOR69-A	1	006212	006373	2	043211	043330
[illegible]/FOR69	1	006374	006434			
[illegible]/FOR68	1	006435	006631	2	043331	043402
[illegible]MONITOR/RFOR69-S	1	006632	010351	2	043403	044350
MAIN	1	010352	010560	0	044351	050326
[illegible]	1	010561	010632	0	050327	050342

SYS$*RLIB$. LEVEL 69-S
END RMAP: 0.530 SECONDS, 031 BLOCKS

A(0) -1.0000
A(1) -.18934
A(2) -.12965
A(3) -.10228
A(4)-8.59658-02
A(5)-7.49254-02
A(6)-6.58656-02
A(7)-6.06754-02
A(8)-5.57424-02
A(9)-5.17021-02
A(10)-4.83205-02
A(11)-4.54412-02
A(12)-4.29543-02
A(13)-4.07804-02
A(14)-3.88511-02
A(15)-3.71316-02
A(16)-3.55176-02
A(17)-3.42319-02
A(18)-3.29727-02
A(19)-3.18225-02
A(20)-3.07672-02
A(21)-2.97946-02
A(22)-2.88949-02
A(23)-2.80597-02
A(24)-2.72819-02
A(25)-2.65555-02
A(26)-2.58755-02
A(27)-2.[illegible]-02

Pareto Model

```
***B(N,K) CALCULATIONS*******

B(  25,2000)=    .000480545699247
B(  26,2000)=    .000467665271572
B(  27,2000)=    .000455601573776
B(  28,2000)=    .000444275337941
B(  29,2000)=    .000433516973169
B(  30,2000)=    .000423565530626
B(  31,2000)=    .000414067875681
B(  32,2000)=    .000405076156312
B(  33,2000)=    .000396549268771
B(  34,2000)=    .000388449214370
B(  35,2000)=    .000380743207643
B(  36,2000)=    .000373401653633
B(  37,2000)=    .000366397383914
B(  38,2000)=    .000359706315066
B(  39,2000)=    .000353306884790
B(  40,2000)=    .000347179124219
B(  41,2000)=    .000341304847097
B(  42,2000)=    .000335667631678
B(  43,2000)=    .000330253489665
B(  44,2000)=    .000325047432852
B(  45,2000)=    .000320037386700
B(  46,2000)=    .000315212004352
B(  47,2000)=    .000310560586513
B(  48,2000)=    .000306073259708
B(  49,2000)=    .000301740619761
B(  50,2000)=    .000297554768622
B(  51,2000)=    .000293507851893
B(  52,2000)=    .000289591993351
B(  53,2000)=    .000285801648804
B(  54,2000)=    .000282129785774
B(  55,2000)=    .000278570831142
B(  56,2000)=    .000275118825812
B(  57,2000)=    .000271769349638
B(  58,2000)=    .000268517331278
B(  59,2000)=    .000265358670731
B(  60,2000)=    .000262288322119
B(  61,2000)=    .000259303320490
B(  62,2000)=    .000256399413047
B(  63,2000)=    .000253573209193
B(  64,2000)=    .000250821311056
B(  65,2000)=    .000248141106567
B(  66,2000)=    .000245529197855
B(  67,2000)=    .000242983074713
B(  68,2000)=    .000240499965003
B(  69,2000)=    .000238077504036
B(  70,2000)=    .000235713532675
B(  71,2000)=    .000233405216932
B(  72,2000)=    .000231151308981
B(  73,2000)=    .000228949060329
B(  74,2000)=    .000226796890274
B(  75,2000)=    .000224693420050
B(  76,2000)=    .000222636144827
B(  77,2000)=    .000220624042413
B(  78,2000)=    .000218655453864
B(  79,2000)=    .000215728454689
B(  80,2000)=    .000214842120867
B(  81,2000)=    .000212994955291
```

Pareto Model Sp(X, 2000)

NF [illegible] . [illegible] 2 042775 043150
043151 043222
NT [illegible] 2 043223 043261
LR [illegible]
[illegible] . 067731 067741
[illegible] . 067742 010120 2 043202 043401
NI VS [illegible] . 010120 010150 2 043402 043492
. I ON [illegible] . 010151 011700 2 043403 044350
RUN . 011701 012000 6 044351 052415

ST [illegible]
FN R [illegible]

SP(1,2000)	=	.67734110169112?-01
SP(2,2000)	=	.67726525890039?-01
SP(3,2000)	=	.677369406654820-01
SP(4,2000)	=	.67633?152340?8?-01
SP(5,2000)	=	.67707??3126583-01
SP(6,2000)	=	.676?61068573594-01
SP(7,2000)	=	.675?247?1575122-01
SP(8,2000)	=	.676?3?32144439?-01
SP(9,2000)	=	.676251?6400?076?-01
SP(10,2000)	=	.676115341404547-01
SP(11,2000)	=	.675?978?3951?561-01
SP(12,2000)	=	.675841?16406641-01
SP(13,2000)	=	.675704?4930107?-01
SP(14,2000)	=	.6755676679313?8-01
SP(15,2000)	=	.67543053068220?-01
SP(16,2000)	=	.675293?4686696?-01
SP(17,2000)	=	.675156004726087-01
SP(18,2000)	=	.675018550828094-01
SP(19,2000)	=	.6748?8945723092?-01
SP(20,2000)	=	.674743354320520-01
SP(21,2000)	=	.674605611711740-01
SP(22,2000)	=	.674467791404891-01
SP(23,2000)	=	.674329?55410623-01
SP(24,2000)	=	.674191778525710-01
SP(25,2000)	=	.674053644701443-01
SP(26,2000)	=	.673915315419430-01
SP(27,2000)	=	.673777059896460-01
SP(28,2000)	=	.673638692229614-01
SP(29,2000)	=	.673500?20036963-01
SP(30,2000)	=	.673361?62306571-01
SP(31,2000)	=	.673222?6040?5534-01
SP(32,2000)	=	.673083?9710?401-01
SP(33,2000)	=	.6729?4?16526365-01
SP(34,2000)	=	.672?0?5930?5652-01
SP(35,2000)	=	.672?6?47703?45?-01
SP(36,2000)	=	.672?26?742?1005-01
SP(37,2000)	=	.672?8?7?4?373842-01
SP(38,2000)	=	.672?48341140131-01
SP(39,2000)	=	.6721?8?3344371?-01
SP(40,2000)	=	.671[illegible]162?5?-01
SP([illegible])	=	.67[illegible]76?717-01
SP([illegible])	=	.67[illegible]2?1?9?-01
SP([illegible])	=	.67[illegible]16?0?96?-01
SP([illegible])	=	.67[illegible]77?4?9?-01
SP([illegible])	=	.67[illegible]2?0?9?7?-01

```
[illegible] LEVEL [illegible]
END MAP: 0.841 SECONDS, 032 BLOCKS
 THE  1000TH INFO SEQUENCE IS 000000017501
 THE  2000TH INFO SEQUENCE IS 000000037201
 AFTER 1-BIT PATTERNS, THERE ARE       2581 OF THEM, THE LAST BEING 000000050251
 THE  [illegible]000TH INFO SEQUENCE IS 000000202662
```

Octal

All octal numbers here will be represented in hexadecimal for IBM 360's output.

Octal

```
 THE  [illegible]000TH INFO SEQUENCE IS 000000702632
 THE  [illegible]000TH INFO SEQUENCE IS 000001601742
 THE  [illegible]000TH INFO SEQUENCE IS 000002701512
 THE  [illegible]000TH INFO SEQUENCE IS 000004301222
 THE  [illegible]000TH INFO SEQUENCE IS 000006300732
 THE  [illegible]000TH INFO SEQUENCE IS 000010701442
 THE  [illegible]000TH INFO SEQUENCE IS 000014201762
 THE  [illegible]000TH INFO SEQUENCE IS 000020702132
 THE  [illegible]000TH INFO SEQUENCE IS 000026703042
 THE  [illegible]000TH INFO SEQUENCE IS 000037103742
 THE  [illegible]000TH INFO SEQUENCE IS 000052205262
 THE  [illegible]000TH INFO SEQUENCE IS 000071607252
 THE  [illegible]000TH INFO SEQUENCE IS 000117712022
 THE  [illegible]000TH INFO SEQUENCE IS 000161116122
 THE  [illegible]000TH INFO SEQUENCE IS 000246024622
 THE  [illegible]000TH INFO SEQUENCE IS 000415741602
 AFTER 2-BIT PATTERNS, THERE ARE     (19421)10 OF THEM, THE LAST BEING 000502450252
 THE  [illegible]000TH INFO SEQUENCE IS 000100420263
 THE  [illegible]000TH INFO SEQUENCE IS 000300240133
 THE  [illegible]000TH INFO SEQUENCE IS 000540300543
 THE  [illegible]000TH INFO SEQUENCE IS 001040520313
 THE  [illegible]000TH INFO SEQUENCE IS 001500660533
 THE  [illegible]000TH INFO SEQUENCE IS 002301220553
 THE  [illegible]000TH INFO SEQUENCE IS 003402021023
 THE  [illegible]7000TH INFO SEQUENCE IS 005302601343
 THE  [illegible]000TH INFO SEQUENCE IS 010604322233
 THE  [illegible]000TH INFO SEQUENCE IS 020610324163
 AFTER 3-BIT PATTERNS, THERE ARE     (29746)10 OF THEM, THE LAST BEING 054126073043
 THE   000TH INFO SEQUENCE IS 002014040124
 THE  [illegible]000TH INFO SEQUENCE IS 016044120254
 THE  [illegible]000TH INFO SEQUENCE IS 115234475214
 AFTER 4-BIT PATTERNS, THERE ARE     (32185) OF THEM, THE LAST BEING 271565356744
 AFTER 5-BIT PATTERNS, THERE ARE     (32362)10 OF THEM, THE LAST BEING 051525354555

NORMAL EXIT.  EXECUTION TIME:        5203 MLSEC.
```

Octal

Octal

D-8

****PROB CALCULATION FOR S554-IRREDUCIBLE

		G(X)	PROB(U.E.)
2nd card	1	40460216667	.969273-11 ←
	2	54114300535	.114413-10
4th card	3	60537314115	.199706-10 ←
3rd card	4	62613476131	.218580-10 ←
	5	52414670717	.218726-10
	6	51474633517	.351764-10
	7	60120240653	.125666-09
	8	51224036761	.844177-08

These 3 polynomials are tested

Octal

APPENDIX III: A NOTE ON THE MUNTER-WOLF CHANNEL MODEL

Care should be exercised in applying the particular case of the Munter and Wolf model discussed on pages 27 and 28 due to the following inconsistency: Combining (1.69) and (1.71) one has

$$P(m,n) = \sum_{i=1}^{M} \lambda_i P_i(1) \frac{\binom{n}{m} \alpha_i^{m+1} K_i^{n+1} (1-\alpha_i)^{n-m}}{[1 - K_i(1 - \alpha_i)]^2} .$$

Noting that

$$\sum_{m=0}^{n} P(m,n) = 1 ,$$

as a fundamental property of P(m,n), implies that

$$\sum_i \lambda_i P_i(1) \frac{\alpha_i K_i^{n+1}}{[1-K_i(1-\alpha_i)]^2} = 1$$

for all n. Hence, $K_i = 1$ and, therefore,

$$\sum_i \lambda_i P_i(1)/\alpha_i = 1.$$

Since $\sum_i \lambda_i = 1$ and from the assumption $P_i(1) << \alpha_i K_i^j$, it is clear that $\sum_i \lambda_i P_i(1)/\alpha_i << 1$. Hence, a contradiction in the model.