C. 1

AUTOMATIC-REPEAT-REQUEST PROTOCOLS FOR DATA COMMUNICATION NETWORKS

by

YET KEUNG CHANG

B.A.Sc, University Of Toronto, 1981

A THESIS SUBMITTED IN PARTIAL FULFILMENT OF

THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF APPLIED SCIENCE

in

THE FACULTY OF GRADUATE STUDIES

Department Of Electrical Engineering

We accept this thesis as conforming

to the required standard

THE UNIVERSITY OF BRITISH COLUMBIA

September 1983

Department of Electrical Engineering

The University of British Columbia
2075 Wesbrook Place
Vancouver, Canada
V6T 1W5

Date: September 20, 1983

Abstract

The performances of a number of Automatic-Repeat-Request (ARQ) protocols are compared on the basis of the expected wasted time per message incurred over random-error and Rayleigh fading channels. These include the standard Stop-And-Wait, Go-Back-N, and Selective-Repeat schemes. The reductions in expected wasted times achievable through the use of forward-error-correction (FEC) are demonstrated. It is found that in general, substantial improvements in performance can be obtained by using FEC.

A new ARQ protocol proposed by Weldon is studied. It is shown that the throughput of Weldon's scheme can be increased by allowing multiple copies of a new data block to be sent. In order to maximize the throughput in Weldon's scheme, a number of parameters need to be selected optimally. An efficient method for choosing these parameters is obtained by exploiting the structure of a simplified expression for the throughput.

The problem of the optimal block length that minimizes the expected wasted time per message is also considered. An exact analysis of the optimal block length is developed for the Stop-And-Wait scheme using an end-of-message character in the last block of a message. The optimal block length is a function of average message length, channel error rate,overhead per packet, acknowledgement delay and transmission rate. It is found that the optimal block length converges to a constant value when the average message length becomes large. Finally, the performance of an algorithm that computes the optimal block length in an

adaptive way is examined.

Table of Contents

List of Tables

List of Figures

Acknowledgement

The author would like to thank Prof. C. Leung for his encouragement and guidance during the course of this thesis.

The Research Assistantship support from the Natural Sciences and Engineering Research Council under Grant NSERC A1052 and A1731 is appreciated.

Special thanks are also extended to B. Maranda for making available results from the Rayleigh fading simulator.

I. INTRODUCTION

1.1 Background

In recent years, the demand for efficient and reliable data communication systems has been greatly accelerated by the rising need for computer-to-computer communication. A serious problem in many data communication systems is the occurrence of errors in the communication channel. Basically, there are two techniques for controlling errors in a data communication system: the Automatic-Repeat-Request (ARQ) scheme and the forward-error-correction (FEC) scheme.

The ARQ scheme can often provide a simple means to achieve high efficiency and reliability for a data communication system. In an ARQ system, the transmitter encodes the message so as to enable the receiver to detect transmission errors and ask for a retransmission of erroneous blocks. ARQ protocols differ in the manner in which the transmitter handles the requests for retransmission. Standard protocols include Stop-And-Wait, Go-Back-N and Selective-Repeat schemes[1,2]. A number of variations of the basic protocols have been suggested[3-6] in order to improve efficiency.

In an FEC system, an error-correcting code is used to correct transmission errors. The receiver attempts to locate and correct the errors in a corrupted packet. Since there is only a certain number of error patterns which can be corrected, the result of the correction may not be the original packet transmitted by the transmitter. Therefore, it may be difficult to achieve high system reliability with FEC. However, for a system in which a return channel is unavailable or retransmission is not possible or convenient, the FEC technique is often used. Hybrid schemes in which FEC is combined with an ARQ protocol are often more efficient than either ARQ or FEC alone[7,8].

In an ARQ system, the message is often split into fixed blocks of size B bits. These are then assembled into packets of length (B+b+p) bits where b represents overhead required for synchronization, addressing, sequence numbering, error detection,etc, and p is the number of parity bits used for error correction. There is a tradeoff involved in selecting the packet size. On the one hand, it is desirable to choose a large packet size so as to reduce the acknowledgement delay and the overhead per message. On the other hand ,a long packet is more likely to be corrupted by the channel,and hence require a retransmission. The problem of how to select the optimal packet length for ARQ protocols has been studied by Chu[9] and Morris[10].

In chapter 2 , a number of ARQ schemes[3-7] are briefly reviewed. Then the performance of various ARQ protocols are compared on the basis of the expected wasted time per message incurred when a message is packetized for retransmission over the channel. The effect of FEC on the expected wasted time of each protocol/channel combination is also examined.

In Chapter 3, a new ARQ scheme proposed by Weldon[6] is studied . A number of methods are proposed to improve the efficiency of Weldon's scheme.

In Chapter 4, the optimal packet length problem is further investigated. Some interesting observations are discussed. An algorithm that computes the optimal packet length in an adaptive way is also developed and its performance is examined by simulation.

1.2 System Model

The system model[6] we are considering consists of a transmitter ,a receiver , a forward data link channel and a noiseless feedback channel.

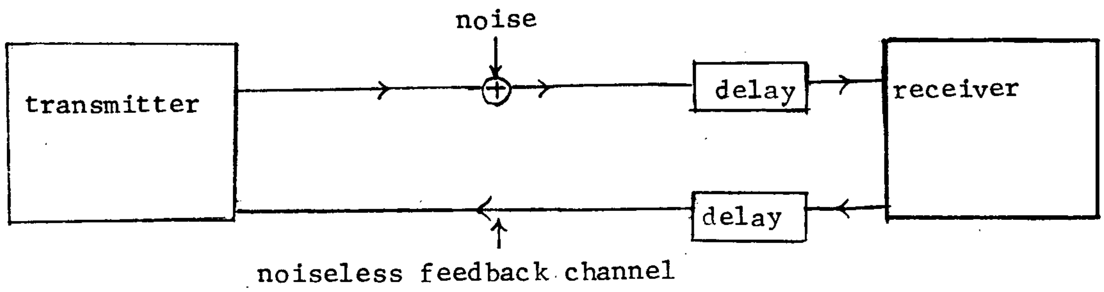

Fig 1.1 The system model.

The major components of the system model are described as follows:

Forward and Feedback channels

The transmitter transmits/retransmits packets to the receiver over the forward data link channel. Packets may be corrupted by noise in the channel. A noiseless feedback channel is provided for the receiver to send positive/negative acknowledgement packets to the transmitter to indicate whether retransmissions are required. The probability of a packet being in error is denoted by P. The number of data packets which can be sent by continuous transmission during the time between the start of the transmission of a packet and the receipt of a positive acknowledgement(ACK) or negative acknowledgement(NACK) for that packet is denoted by S. The throughput efficiency T is defined as the ratio of the time taken to transmit a packet assuming the channel is noiseless to the average time involved in the transmission of the same packet over the actual channel. Note that our definition of throughput does not take overhead bits e.g. for addressing , error control etc. into account. However , the effects of these overhead bits will be considered in Chapter 2.

Transmitter

Each message arriving at the transmitter is split into blocks of size B bits. These are then assembled into packets of length (B+b+p) bits where b represents overhead required for synchronization, addressing,sequence numbering, error detection ,etc. and p is the number of parity check bits if an error correcting code is used. The transmitter then proceeds to transmit or retransmit the packet over the channel and waits for the positive/negative acknowledgement from the receiver.Since all the corrupted packets have to be retransmitted, the transmitter has a buffer size of at least S blocks to store each transmitted packet until it is positively acknowledged by the receiver.

Receiver

Upon receiving a packet , the receiver proceeds to decode the packet. It is assumed that all errors in the received packet are detected at the receiver[11]. If an error-correcting code is used, the receiver will attempt to correct the corrupted packet. Whenever a packet is received correctly or error correction is successful, an ACK is returned to the transmitter. However, if an uncorrectable error pattern occurs , the result of the correction is not the original packet transmitted by the transmitter. Upon detecting the errors , the receiver requests a retransmission by sending a NACK to the transmitter. In order to store the received packets, the receiver has a buffer size of 1 block or qS blocks, q=1,2,3,..., depending on the particular ARQ scheme in use. For the class of Selective-Repeat protocols, the receiver buffer stores all the correct packets following the bad one until the buffer is full. Once a bad packet is recovered by retransmission , all the <u>consecutive correct packets</u> following the recovered packet are released and subsequently delivered to the host computer(data sink). Therefore the head of the buffer queue is always a corrupted packet followed by some correct/corrupted packets. In this way, all the packets received by the host computer are in consecutive order. This measure is for the ease of main memory management in the host computer.

II. PERFORMANCE COMPARISON OF SOME ARQ SCHEMES

2.1 Review Of Some Specific ARQ Schemes

ARQ schemes are commonly used in data communication systems because they can often provide a simple means for achieving high efficiency and reliability. However, ARQ schemes have a common shortcoming: the throughputs of ARQ schemes decrease quite rapidly with increasing error probability. For satellite communication applications , the presence of large round trip delays aggravates this problem. Therefore, much effort has been devoted to the study of ARQ schemes that maintain high throughput efficiency for high error rate and large round trip delay channels. In this section, a number of representative ARQ schemes to be compared in the next section are briefly reviewed. These include the standard Stop-And-Wait ,Go-Back-N and Selective-Repeat schemes[1,2] as well as a number of variations suggested[3-6] by some authors.

The throughput of these ARQ schemes are now listed as follows:

(1) Stop-And-Wait

After sending a packet, the transmitter waits for an ACK or NACK from the the receiver to determine whether to transmit a new packet or retransmit the same packet. The throughtput is given by

$$T = \frac{(1-P)}{S}. \qquad (2.1.1)$$

In the Stop-And-Wait scheme ,the transmitter is often idle. In all the following protocols, the transmitter is always sending either new packets or retransmissions. For this reason, these protocols are referred to as continuous ARQ.

(2) Basic Go-Back-N (GBN)

The transmitter sends new packets continuously until a NACK is received. Then the transmitter retransmits the NACKed packets, followed by all subsequent(possibly previously transmitted) packets. Note that the receiver requires only a buffer size of one packet. The throughput T is given by

$$T = \frac{(1-P)}{1 + (S-1)P} \,. \qquad (2.1.2)$$

(3) Sastry's GBN [3]

In this modification of the basic Go-Back-N protocol, the transmitter enters a stutter mode as soon as a NACK for a packet is received, i.e., the transmitter keeps sending the NACKed packet(say i) continuously until an ACK for packet i is received. At this point, the transmissions of packets i+1,i+2, etc. are resumed. For this protocol,

$$T = \frac{1-P}{1 + 2P(1-P)(S-1)} \,. \qquad (2.1.3)$$

(4) Morris' GBN [4]

This is an enhancement of Sastry's GBN protocol with a receiver buffer of size S packets to save the good packets following a corrupted one. Here the throughput is increased to

$$T = \frac{1-P}{1 + P(1-P)(S-1)} \qquad (2.1.4)$$

(5) Selective-Repeat plus Stutter I (SR+ST) [5]

This protocol is an extension of Morris' GBN scheme in that a buffer of size qS, q=1,2,3,.... is available at the receiver. For the first (q-1) times that a given packet is NACKed , the transmitter retransmits it using a selective repeat mode. If a qth NACK for that packet is received, the transmitter switches to a stutter mode. The throughput is given by

$$T = \frac{1-P}{1 + P^q(1-P)(S-1)} \,. \qquad (2.1.5)$$

In chapter 3, we will show that this protocol can be extended along the lines of Weldon scheme discussed below.

(6) Weldon's scheme [6]

In this protocol, the receiver has a buffer of size qS packets,where q=1,2,3,.... . At any given point in time, a packet B can be in any level i, $0 \leq i \leq q+1$. Initially, B is at level 0. At any given level j, B is repeated n_j times , where n_j is a parameter to be chosen. If any of these n_j copies is

ACKed, the transmission of B is complete. If all n_j copies are NACKed, B moves to level j+1. In this case the throughput T is given by $1/\beta$ [6] where

$$\beta = \sum_{i=0}^{q} n_i \, P^{\sum_{k=0}^{i-1} n_k} + \frac{(n_q+S-1)P^{\sum_{k=0}^{q} n_k}}{1-P^{n_q}} . \qquad (2.1.6)$$

A more detailed description of Weldon's scheme [6] is given in Appendix A. It might be noted that many previously suggested ARQ protocols are special cases of Weldon's ARQ scheme. For example, the classical Selective-Repeat protocol is obtained by setting q=1 and $n_0=n_1=1$, and results in a throughput of

$$T = \frac{1-P}{1+(S-1)P^2} . \qquad (2.1.7)$$

The Selective-Repeat plus Go-Back-N (SR+GBN) protocol of [5] is obtained by setting $n_0=n_1=....=n_q=1$, and yields a throughput of

$$T = \frac{1-P}{1+(S-1)P^{q+1}} . \qquad (2.1.8)$$

(7) IDEAL SELECTIVE-REPEAT

In this protocol , a retransmission for a packet is made only if all previous copies of that packet are erroneous. This scheme requires a buffer of infinite size at the receiver and is therefore not very practical. Nevertheless it is useful as a basis for comparing other ARQ schemes. The throughtput is given by

$$T = 1-P. \tag{2.1.9}$$

2.2 Effect Of Forward Error Correction On Expected Wasted Time

2.2.1 Expected Wasted Time Analysis

In this section , the throughput expressions given in Section 2.1 are used to obtain expressions for the expected wasted times of the various ARQ protocols, both with and without FEC. Two channel models, one with random errors and the other with Rayleigh fading, are considered. The model used for determining the expected wasted time is the one presented in [9]. Messages are assumed to have geometrically distributed random lengths, L,[21].i.e.

$$P_L(\ell)=(1-q)q^{\ell-1} \quad ,\ell=1,2,3,.. \tag{2.2.1}$$

with average length $\overline{L}=1/(1-q)$. Each message is split into blocks of size B bits. The blocks are then assembled into packets of length $n=(B+b+p)$ bits where b represents overhead required for synchronization, addressing, error detection, etc. and p is the number of parity bits used for error correction.

We now proceed to calculate the expected wasted time per message. The wasted time is defined to be the difference between the actual time required for transmitting the packetized message and the time it would take to directly transmit the unpacketized message over an error-free channel with the same bit rate. Assuming that errors occuring in different packets are independent and that the last packet is filled with (if necessary) dummy bits to make all packets of fixed length, the expected wasted time per message [8,9] is given by

$$\overline{W} = \overline{N}(B) \cdot \frac{1}{T} \cdot D - \frac{\overline{L}+b}{R} \tag{2.2.2}$$

where

R=transmission rate in bits/sec.

$\overline{N}(B)$=average number of packets per message

D=n/R=time to transmit a packet in sec.

T=throughput of the ARQ scheme employed as given in section 2.1

For geometrically distributed message lengths, the mean number of packets per message is given by

$$\begin{aligned}\overline{N}(B) &= \sum_{n=1}^{\infty} n \cdot P((n-1)B < L \leq nB) \\ &= \sum_{n=1}^{\infty} n \sum_{\ell=(n-1)B+1}^{nB} (1-q)\, q^{\ell-1} \\ &= \frac{1}{1-q^{B}}\end{aligned} \tag{2.2.3}$$

Therefore, the expected wasted time per message for geometrically distributed message length is:

$$\overline{W} = \frac{1}{(1-q^{B})} \cdot \frac{1}{T} \cdot D - \frac{\overline{L} + b}{R} \qquad (2.2.4)$$

We now consider the effect of FEC on the expected wasted time as described by equation (2.2.4). It should be noted that there is a tradeoff involved in using an error-correcting code: on the negative side, error correction parity check bits introduce additional overhead in a packet . However, this may be more than offset by the reduction in the probability of a packet retransmission P. The error-correcting codes we are considering are the Bose-Chaudhuri-Hocquenghem (BCH) codes[12,13]. Given a packet length n (preferably of the form $n=2^{m}-1$, m=2,3,4,... corresponding to the codeword lengths of BCH codes),the distribution of the number of channel errors in the packet is obtained. This can be derived analytically for the random-error channel. For the Rayleigh fading channel, simulation [14,15] were used to obtain the desired distribution. From this distribution , the probability of a block being in error P(n,t),which results when a t-error-correcting code is used,can be determined. P(n,t) then allows the determination of the throughput T for use in evaluating the expected wasted time in equation (2.2.4). The corresponding number of parity check bits p required can be easily obtained from BCH code parameters(n,k,t), namely:

$$n=2^{m}-1$$

$$k=B+b\geqslant n-mt$$

$$p=n-k$$

The resulting expression for the expected wasted time can be obtained by comparing with equation (2.2.4) and is given by

$$\overline{W}=\frac{1}{1-q^{n-b-p}}\cdot\frac{1}{T}\cdot D-\frac{\overline{L}+b}{R} \qquad (2.2.5)$$

For a given packet length n, equation (2.2.5) is evaluated for different values of t to determine the choice of t which minimizes $\overline{W}$.

2.2.2 Numerical Results For The Random-error Channel

The distribution of the number of bit(channel) errors in a packet of length n transmitted over a random-error channel is given by the binomial distribution

$$P_{random}(n,N)=\binom{n}{N}p_b^{\ N}(1-p_b)^{n-N} \qquad (2.2.6)$$

where P_b is the bit error rate.

If no error correcting code is used, the probability of a block being in error P is given as

$$P=1-P_{random}(B+b,0) \qquad (2.2.7)$$

If a t-error-correcting code is used, the probability of a block being in error P is given as

$$P=P(n,t)=1-\sum_{i=0}^{t}P_{random}(n,i) \qquad (2.2.8)$$

An idea of how the number, t, of errors corrected affects the expected wasted time per message $\overline{W}$ can be obtained from Figures

(2.1) ,(2.2) and (2.3). These are plots of $\overline{W}$ as a function of t for the Stop-And-Wait,Go-Back-N and Selective-Repeat schemes respectively.The parameter A is the acknowledgement delay in seconds given by (S-1)D. The plots show that for a given blocklength n,there is an optimum value for t. Moreover, $\overline{W}$ is quite sensitive to t, especially for the smaller values of n.

The expected wasted time $\overline{W}$ is plotted as a function of B,the number of information bits per packet, for 7 different ARQ schemes with and without error correction in Fig (2.4) to (2.7). In order to provide a common basis for comparison, a receiver buffer size of S packets is assumed(i.e.,q=1) for the SR+GBN, SR+ST and Weldon schemes.Hence in these figures, the curves for SR+GBN and Morris are the same as those for the classical SR and SR+ST respectively. The parameters used in Figure (2.7) might correspond to a satellite system with large file transfers. The solid curves corresponding to the use of FEC are obtained by using the optimum t for values of n of the form 2^m-1. In each figure , there is an optimal value of B which minimizes the expected wasted time for each ARQ scheme as suggested by the discussion above.

From these figures, it is clear that FEC can substantially reduce $\overline{W}$. It is also interesting to note that FEC tends to equalize the expected wasted times of the continuous ARQ protocols. This can be explained by the fact that the use of FEC effectively results in a channel with a lower block error probability P. As P decreases, the difference in throughputs between the continuous ARQ schemes becomes smaller .

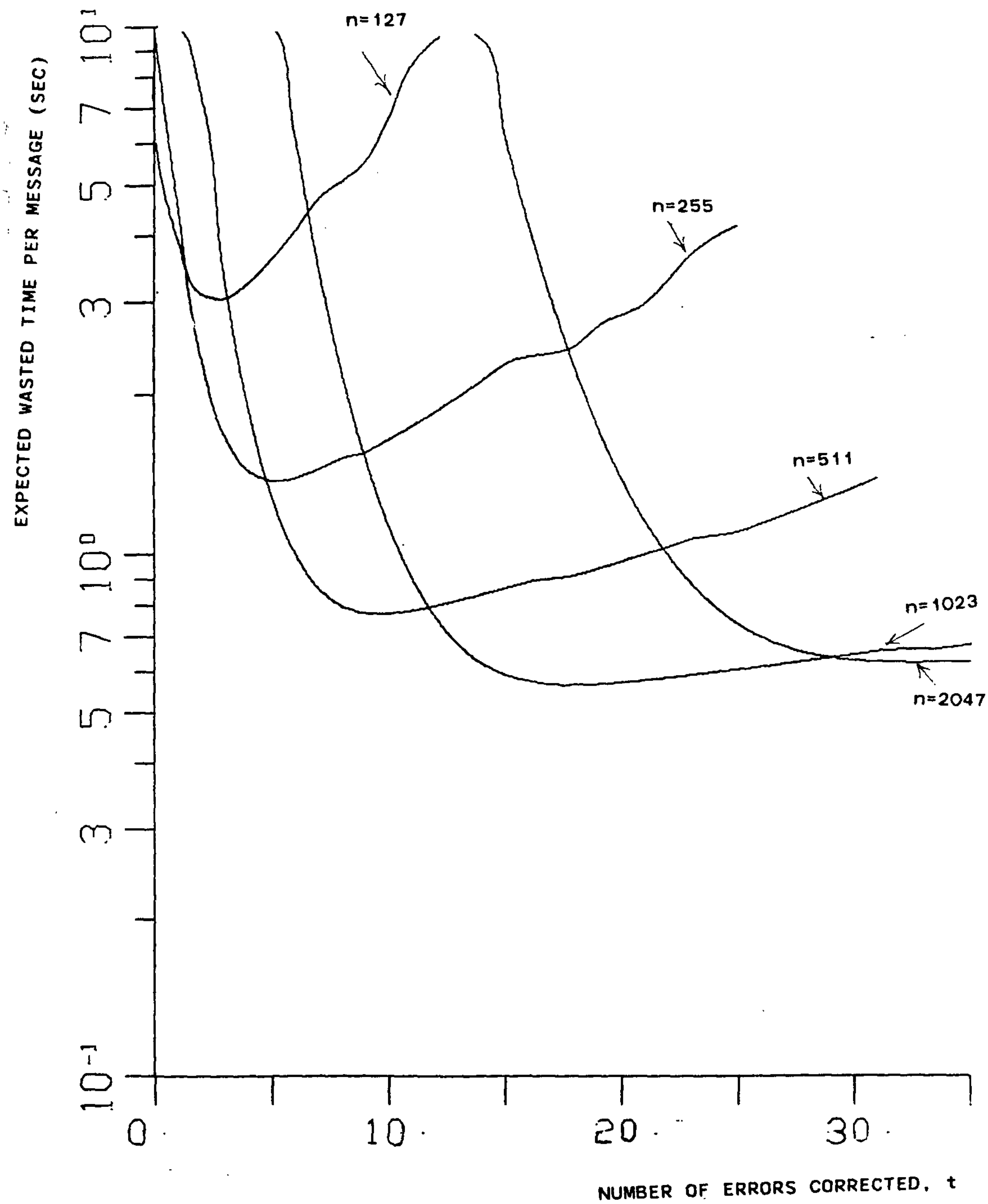

Fig 2.1-Expected wasted time against number of errors corrected (with packet length n as a parameter)for Stop-And-Wait scheme in random error channels with P_b =0.01,R=4000bits/s, $\overline{L}$=1000 bits, A=0.2s, and b=30 bits.

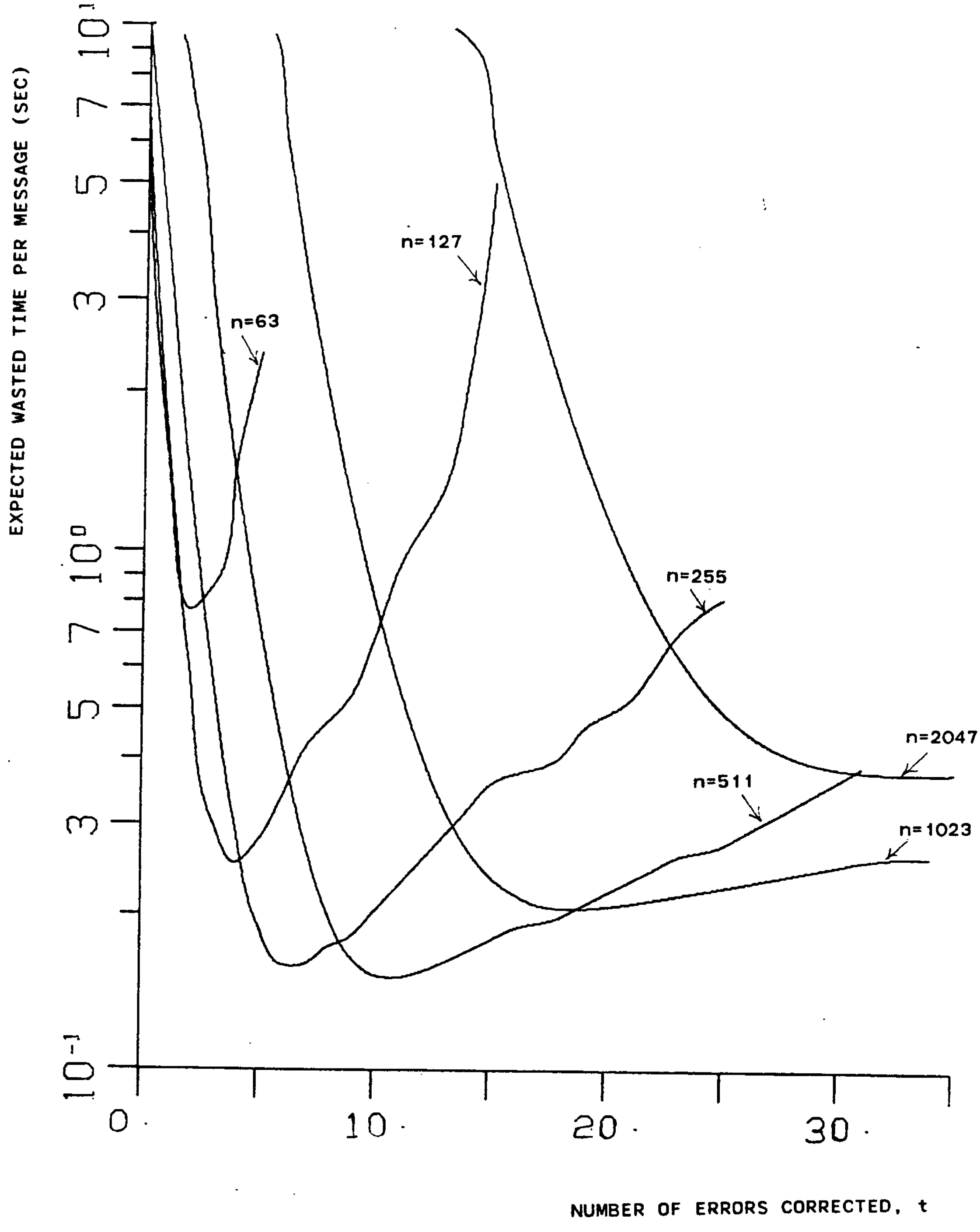

Fig 2.2-Expected wasted time against number of errors corrected (with packet length n as a parameter)for Go-Back-N scheme in random error channels with P_b=0.01,R=4000bits/s, $\overline{L}$=1000 bits, A=0.2s, and b=30 bits.

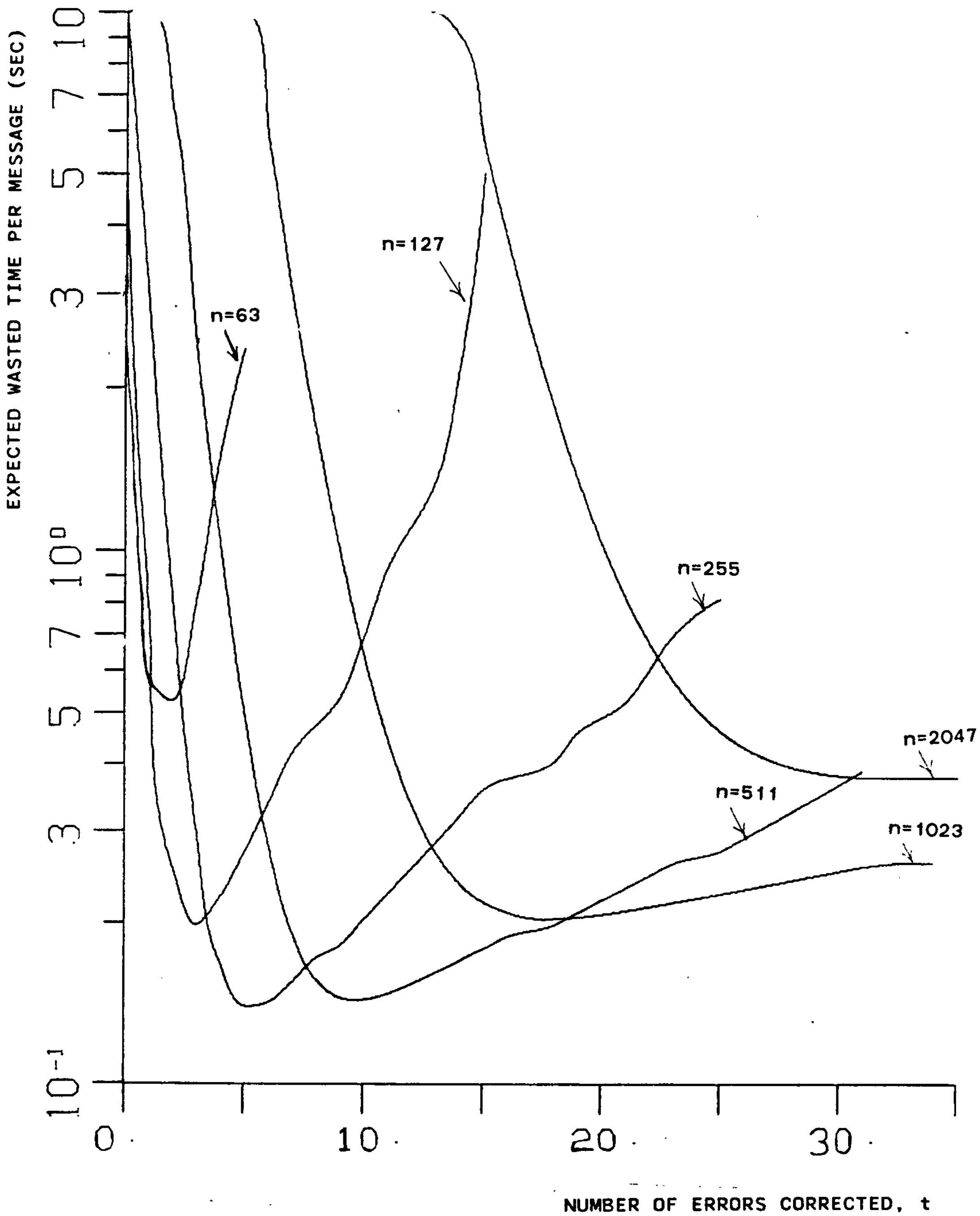

Fig 2.3-Expected wasted time against number of errors corrected (with packet length n as a parameter)for Selective-Repeat (q=1) scheme in random error channels with P_b =0.01, R=4000 bits/s, $\overline{L}$=1000 bits, A=0.2s, and b=30 bits.

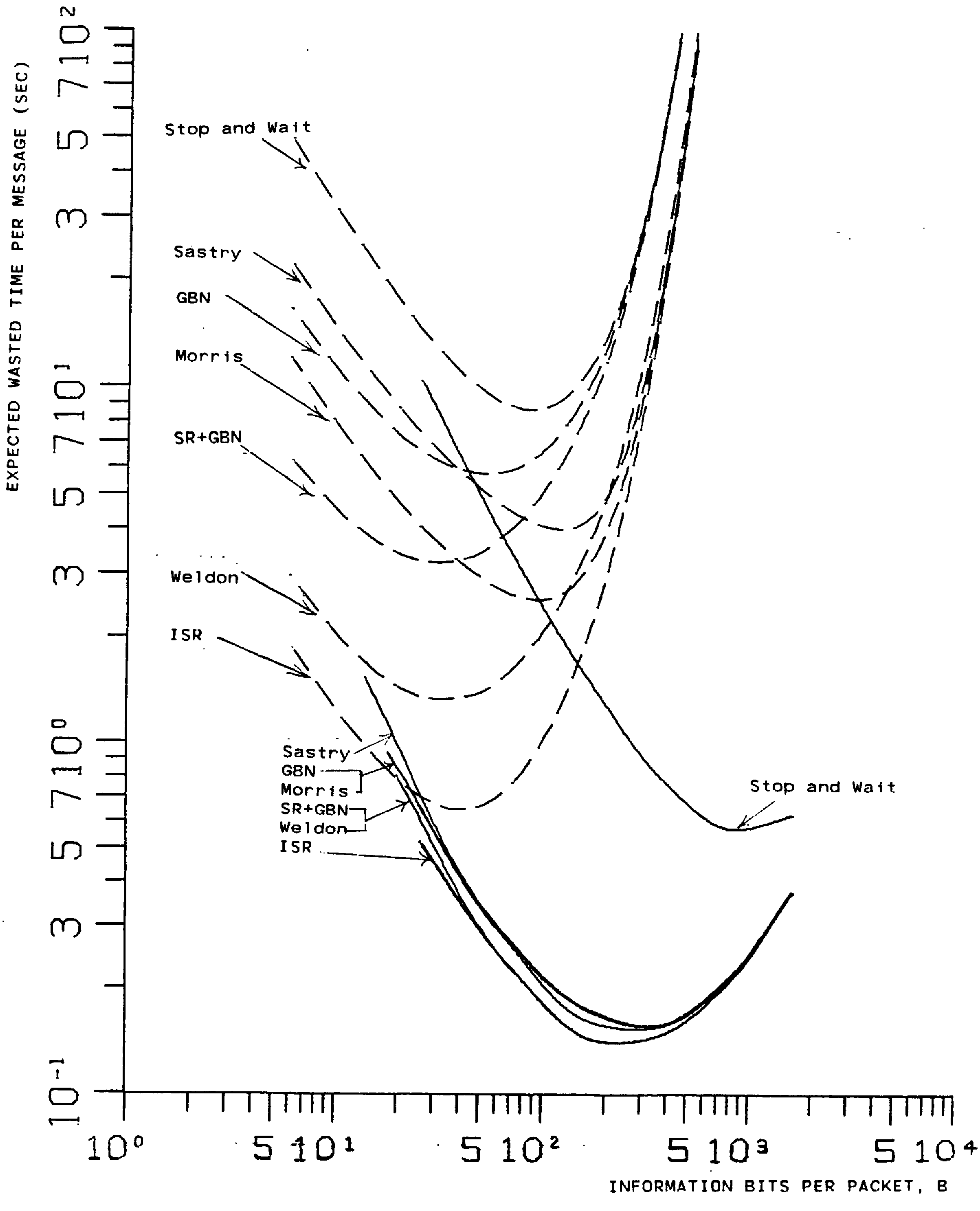

Fig 2.4-Expected wasted time against B for random error channel with P_b =0.01,R=4000bits/s,$\overline{L}$=1000bits,A=0.2s,and b=30bits. Solid Lines denote using error correction. Dashed lines denote without using error correction.

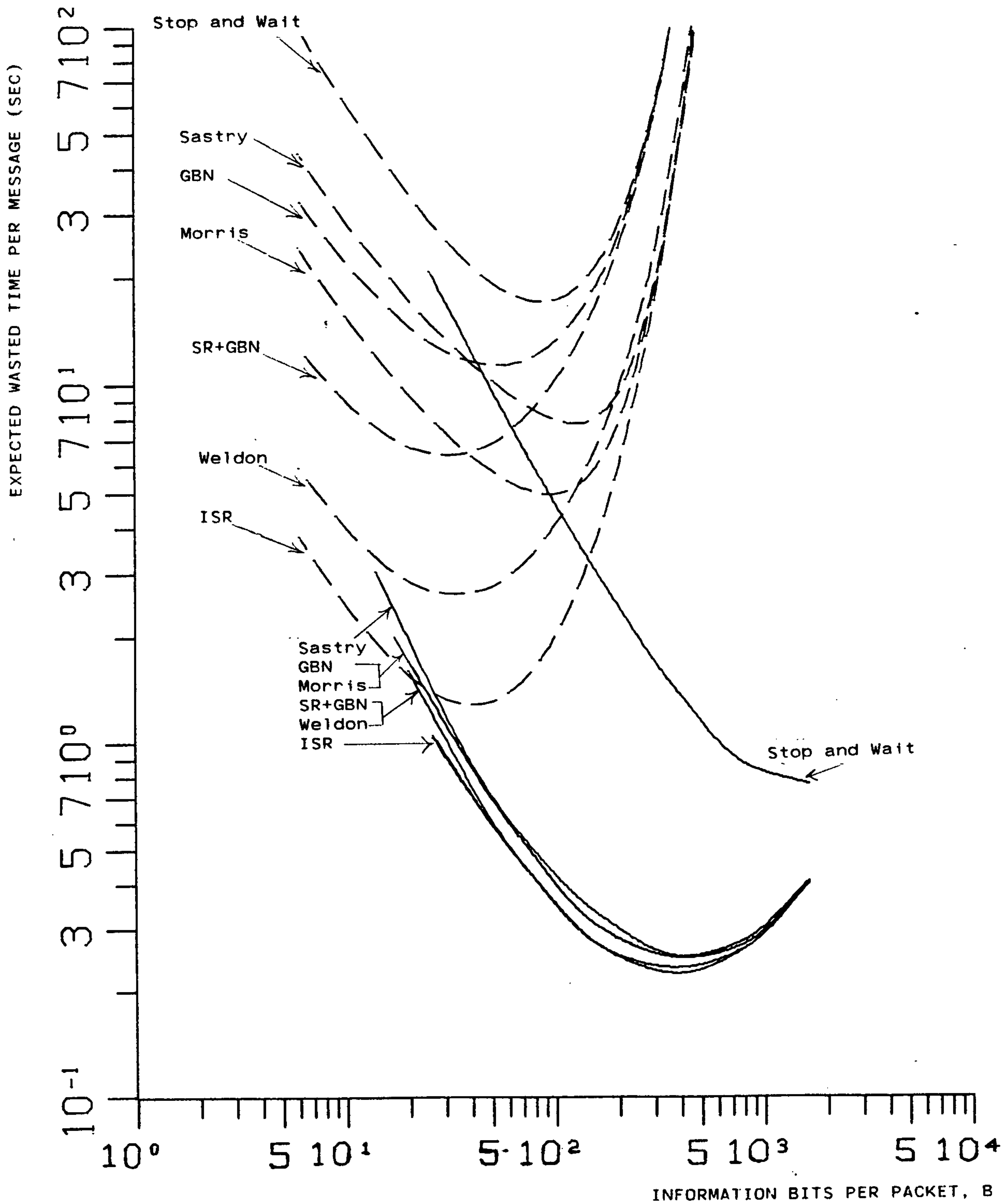

Fig 2.5-Expected wasted time against B for random error channel with $P_b=0.01$, $R=4000$ bits/s, $\overline{L}=2000$bits, $A=0.2$ s, and $b=30$ bits. Solid Lines denote using error correction. Dashed lines denote without using error correction.

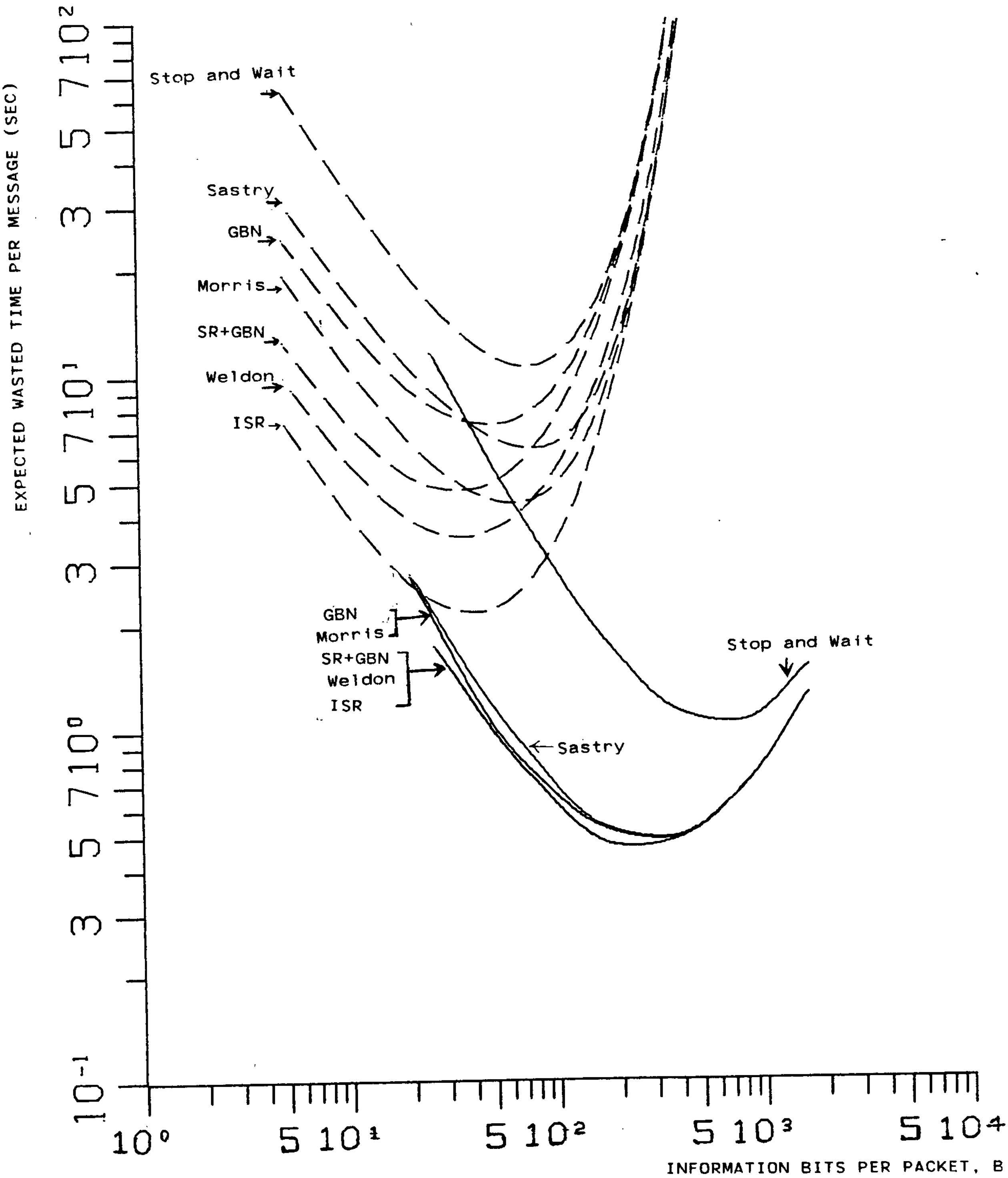

Fig 2.6-Expected wasted time against B for random error channel with P_b=0.01,R=1200 bits/s,$\overline{L}$=1000bits,A=0.2s,and b=30bits. Solid Lines denote using error correction. Dashed lines denote without using error correction.

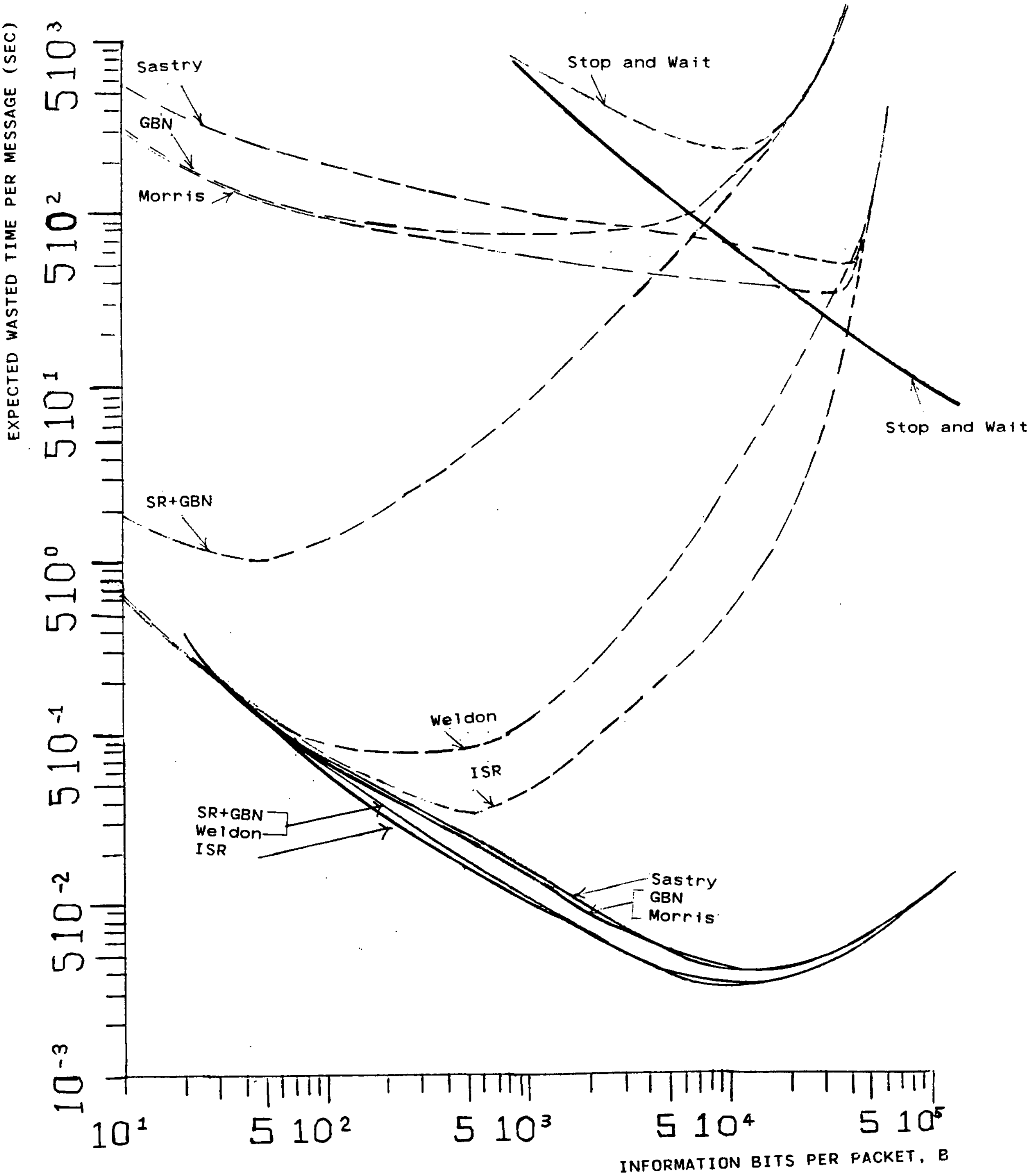

Fig 2.7-Expected wasted time against B for random error channel with P_b=0.0001,R=5000000bits/s,$\overline{L}$=1000000bits,A=0.7s,and b=30 bits. Solid Lines denote using error correction. Dashed lines denote without using error correction.

2.2.3 Numerical Results For Rayleigh Fading Channel

The Rayleigh fading channel is commonly used to model the transmission of data over mobile VHF/UHF radio channels. The distribution of the number of bit errors can be obtained from a simulation program[14,15].A typical distribution is given in Fig (2.8), which shows the cumulative distribution function (CDF) of the number of errors in packets of length 63,127,255,511,1023 and 2047 bits. In this figure, a bit error rate P_b of 0.01 ,a transmission rate R of 4000 bits/sec, a Doppler frequency f_D of 25 HZ(corresponding to a vehicle speed of about 20 MPH at a carrier frequency of 850 MH) and non-coherent FSK modulation were assumed.

The results from Fig (2.8) can be used in conjunction with equation (2.2.4) and (2.2.5) to obtain Fig (2.9). As in the random error channel case, the curves with FEC correspond to the use of the optimum values of t. Here again, it can be seen that FEC can substantially reduce the expected wasted time $\overrightarrow{W}$. However, the reduction are not as large as for the random-error channel.

2.2.4 Concluding Remarks

It has been shown that FEC can be used to substantially improve the performance of a number of different ARQ schemes. The improvement in a given application will obviously depend on system parameters such as packet retransmission probability,the acknowledgement delay, etc. and on the distribution of channel errors. The benefit of any reduction in wasted time will have to be weighed against the additional cost and complexity of using FEC.

The expected wasted time can be quite sensitive to the number, t ,of errors corrected(assuming FEC is used). Hence, the choice of t should be made with some care. It is also possible that the use of FEC will influence the choice of the particular ARQ scheme used. For a given set of system parameters such as transmission rate,acknowledgement delay , overhead per packet, etc. , the improvement in expected wasted time by the use of FEC depends on the channel bit error rate P_b .One would expect the improvement to decrease with P_b . Table (2.1) to (2.2) show the expected wasted times with no error correction and with error correction as a function of P_b for the Go-Back-N and Weldon(assuming q=1) schemes.The expected wasted times for no error correction are optimized with respect to packet lengths n of the form 2^m-1 , and the expected wasted times with error correction are optimized with respect to n and the class of BCH codes. Thus, for the particular example considered, the use of error correction is quite beneficial for $P_b > 0.0005$. Below $P_b \cong 0.0001$, there is little gain that is obtained. This can be illustrated in Table (2.2) in which the optimum number of errors to be corrected is zero for $P_b \lesssim 0.0001$ for Weldon's scheme(q=1).

TABLE 2.1

EXPECTED WASTED TIME PER MESSAGE FOR RANDOM-ERROR CHANNEL WITH $R=4000$BITS/SEC,$\overline{L}=1000$BITS,$A=0.2$SEC,$b=30$BITS FOR GO-BACK-N SCHEME

P_b	Expected Wasted Time (sec)		BCH Codes used (n,k,t)
	no error correction	with error correction	
0.01	6.10	1.51	(511,412,11)
0.005	2.32	0.12	(255,233, 4)
0.001	0.42	0.08	(255,239, 2)
0.0005	0.237	0.079	(255,247, 1)
0.0001	0.093	0.070	(255,247, 1)
0.00005	0.076	0.069	(255,247, 1)
0.00001	0.062	0.062	(255,255, 0)

TABLE 2.2

EXPECTED WASTED TIME PER MESSAGE FOR RANDOM-ERROR CHANNEL WITH $R=4000$BITS/SEC,$\overline{L}=1000$BITS,$A=0.2$SEC,$b=30$BITS FOR WELDON'S SCHEME WITH $q=1$.

P_b	Expected Wasted Time (sec)		BCH Codes used (n,k,t)
	no error correction	with error correction	
0.01	1.32	0.14	(255,215, 5)
0.005	0.643	0.109	(255,231, 3)
0.001	0.167	0.079	(255,247, 1)
0.0005	0.117	0.071	(255,247, 1)
0.0001	0.068	0.067	(255,255, 0)
0.00005	0.063	0.063	(255,255, 0)
0.00001	0.060	0.060	(255,255, 0)

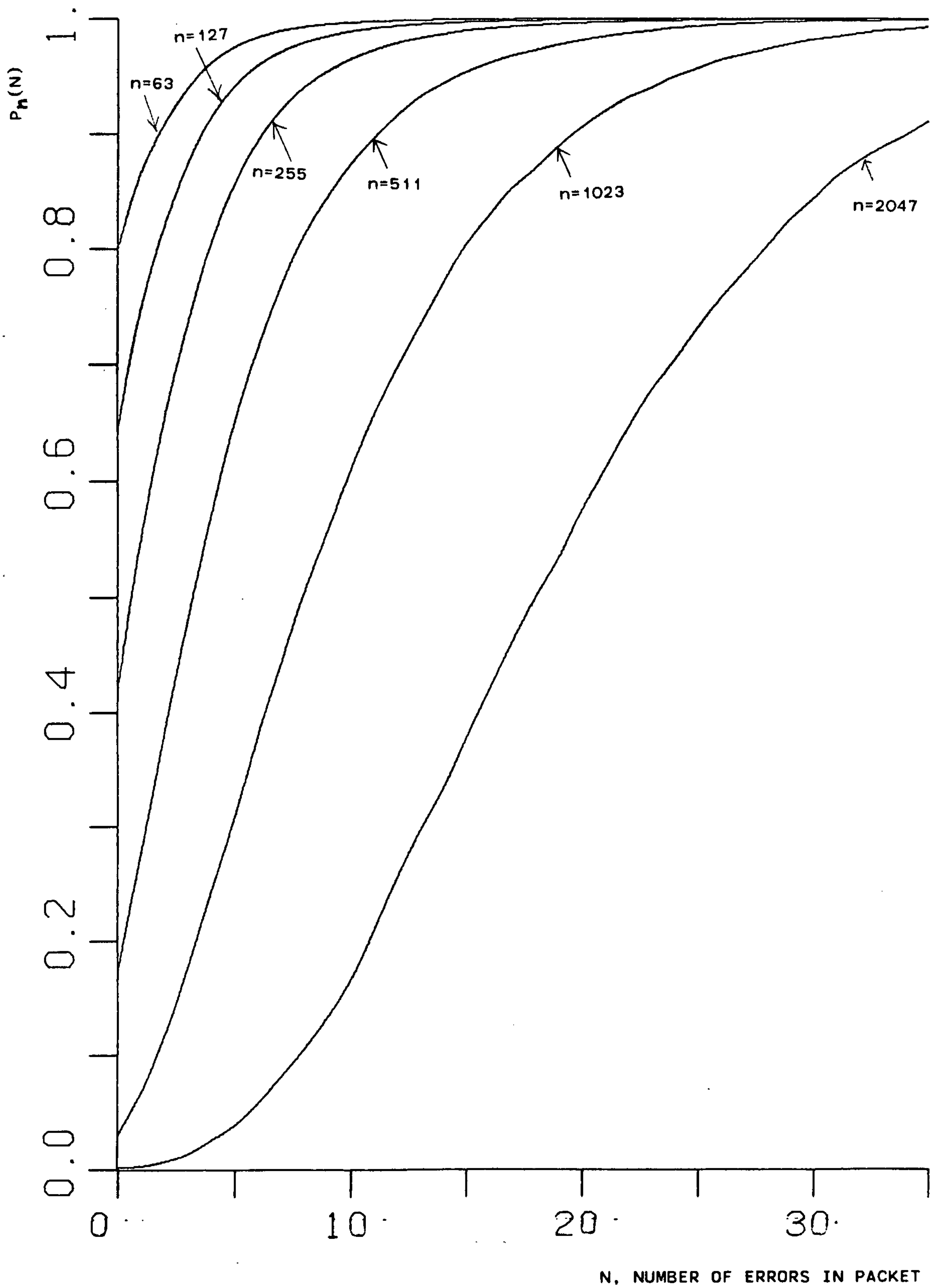

Fig 2.8- CDF, $P_n(N)$ of number of errors in packets of lengths n for Rayleigh fading channel with $P_b = 0.01$, $f_D = 25$ Hz, and transmission rate R=4000 bits/s.

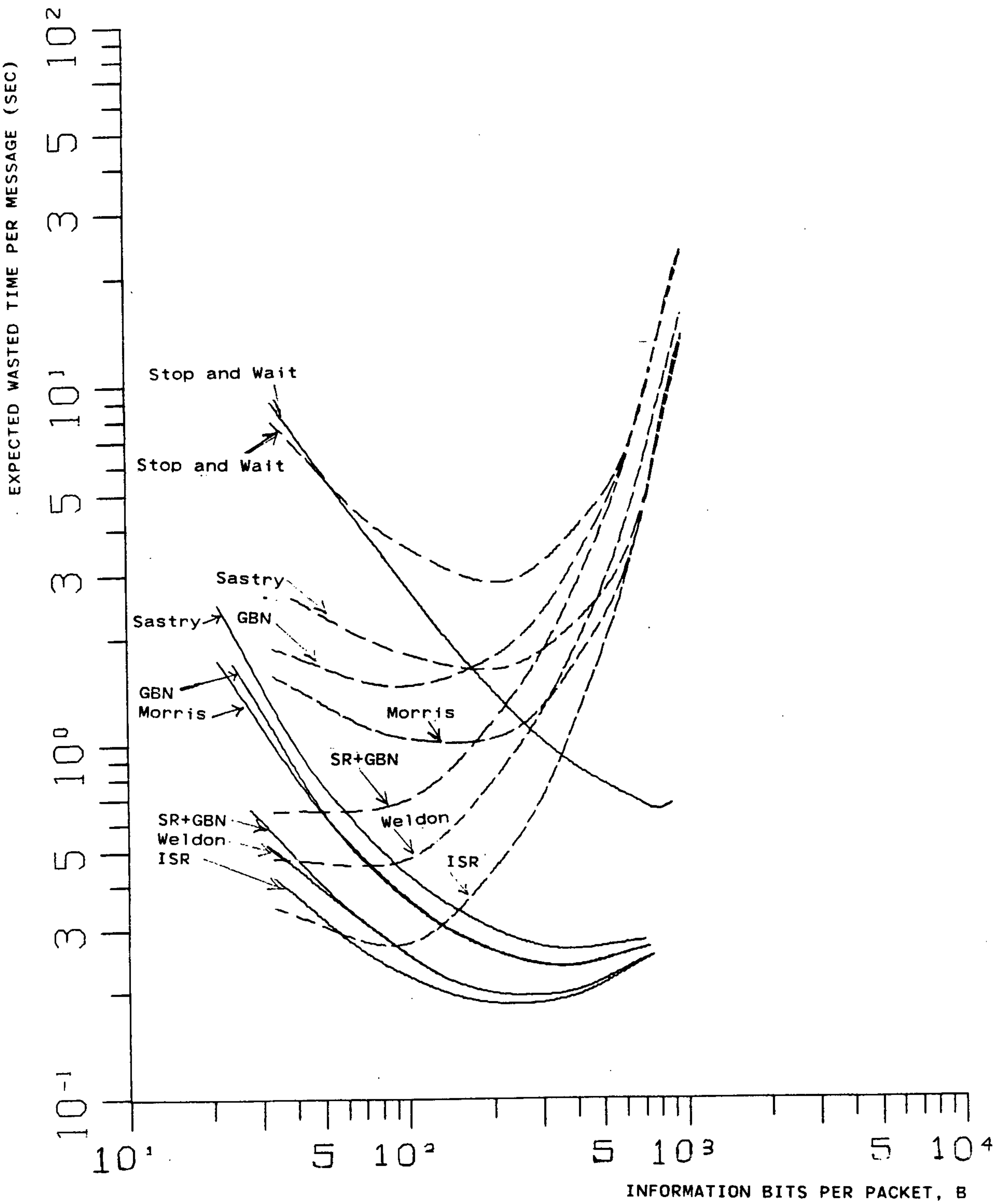

Fig 2.9-Expected wasted time against B for Rayleigh fading channel with P_b =0.01,R=4000bits/s,$\overline{L}$=1000bits,A=0.2s,and b=30 bits. Solid Lines denote using error correction. Dashed lines denote without using error correction.

III. ON WELDON'S ARQ SCHEME

3.1 Weldon's ARQ Scheme

Recently, Weldon has proposed a new ARQ protocol [6] which appears to have a higher throughput than any of the previously known practical schemes (naturally, the Ideal Selective Repeat protocol which requires an infinite buffer at the receiver has the highest throughput: 1-P). The merit of Weldon's scheme is in the idea of repeating NACKed packets multiple times with the number of repeats increasing as the receiver buffer approaches overflow. Moreover,the number of repeats is optimized to yield the highest throughput performance. It might be noted that many of the wellknown ARQ schemes such as GBN, Selective Repeat , etc. are special cases of Weldon's scheme.

In this chapter, some methods to improve Weldon's scheme are discussed. First, a simplification of the throughput expression of Weldon's scheme is presented. Then it is shown that the throughput of Weldon's scheme can be increased by allowing multiple copies of a new data block to be sent.Also, by exploiting the structure of the simplified throughput expression , an efficient method for determining the optimum number of repeats $\{n_i\}$ can be obtained. Finally a modified Weldon ARQ scheme that prevents receiver buffer overflow is presented, and its throughput performance is analyzed. A description of Weldon's scheme [6] is included in Appendix A.

3.2 A Simplified Throughput Expression For Weldon's ARQ Scheme

Following [6], we define $\beta = 1/T$ as the average number of transmissions required to successfully send one block. Equation (8) of [6] shows that

$$\beta = \sum_{i=0}^{q} \left\{ \sum_{j=0}^{i} n_j \right\} P^{\sum_{k=0}^{i-1} n_k} (1 - P^{n_i})$$

$$+ \sum_{i=0}^{q} n_i \, P^{\sum_{k=0}^{q} n_k} + \frac{(n_q + S - 1) P^{\sum_{k=0}^{q} n_k}}{1 - P^{n_q}} . \qquad (3.2.1)$$

It is found that the above expression can be simplified to

$$\beta = \sum_{i=0}^{q} n_i \, P^{\sum_{k=0}^{i-1} n_k} + \frac{(n_q + S - 1) \, P^{\sum_{k=0}^{q} n_k}}{1 - P^{n_q}} \qquad (3.2.2)$$

The simplification is done by noting that the first two sums of equation (3.2.1) can be reduced to

$$\sum_{i=0}^{q}\Big(\sum_{j=0}^{i} n_j\Big) P^{\sum_{k=0}^{i-1} n_k}\big(1-P^{n_i}\big) + \sum_{i=0}^{q} n_i P^{\sum_{k=0}^{q} n_k} = \sum_{i=0}^{q} n_i P^{\sum_{k=0}^{i-1} n_k} \qquad (3.2.3)$$

The L.H.S. of equation (3.2.3) can be re-written as

$$\sum_{i=0}^{q}\Big(\sum_{j=0}^{i} n_j\Big) P^{\sum_{k=0}^{i-1} n_k} - \sum_{i=0}^{q}\Big(\sum_{j=0}^{i} n_j\Big) P^{\sum_{k=0}^{i} n_k} + \sum_{i=0}^{q} n_i P^{\sum_{k=0}^{q} n_k} = \sum_{i=0}^{q} n_i P^{\sum_{k=0}^{i-1} n_k} + F$$

where

$$F \triangleq \sum_{i=0}^{q}\Big(\sum_{j=0}^{i-1} n_j\Big) P^{\sum_{k=0}^{i-1} n_k} - \sum_{i=0}^{q}\Big(\sum_{j=0}^{i} n_j\Big) P^{\sum_{k=0}^{i} n_k} + \sum_{i=0}^{q} n_i P^{\sum_{k=0}^{q} n_k}. \qquad (3.2.4)$$

Expanding the second term in F , we have

$$\begin{aligned} F &= \sum_{i=0}^{q}\Big(\sum_{j=0}^{i-1} n_j\Big) P^{\sum_{k=0}^{i-1} n_k} - \Big[\sum_{i=0}^{q-1}\Big(\sum_{j=0}^{i} n_j\Big) P^{\sum_{k=0}^{i} n_k} + \sum_{j=0}^{q} n_j\, P^{\sum_{k=0}^{q} n_k}\Big] + \sum_{i=0}^{q} n_i P^{\sum_{k=0}^{q} n_k} \\ &= \sum_{i=0}^{q}\Big(\sum_{j=0}^{i-1} n_j\Big) P^{\sum_{k=0}^{i-1} n_k} - \sum_{i=0}^{q-1}\Big(\sum_{j=0}^{i} n_j\Big) P^{\sum_{k=0}^{i} n_k} \\ &= 0. \end{aligned} \qquad (3.2.5)$$

This completes the proof of equation (3.2.2).

3.3 Weldon's ARQ Scheme With Variable n_0

For given values of S and P, the parameters $\{n_i\}_{i=0}^{q}$ can be chosen so as to minimize β (or equivalently maximize the throughput T). In Weldon's original scheme [6], the parameter n_0 which is the number of transmissions of a block at level 0, is set to 1.However , this may not be the optimum value for n_0 ,as indicated in Figure (3.1). Here the throughput is plotted against the block error rate P for S=1000 blocks and q=1, an example used in [6]. For $P \lesssim 0.25$, the optimum value of n_0 is 1. However, for larger values of P ,the throughput is increased by using larger values of n_0.In section 3.4, a detailed analysis of the optimum values of $\{n_i\}_{i=0}^{q}$ is given . At this point, it suffices to note that the Weldon scheme with variable n_0 can have a significantly higher throughput . As an example, for P=0.6 , Figure (3.1) shows that T can be increased from 0.10 to 0.17. Also plotted in Figure (3.1) is the throughput curve for the ideal selective repeat scheme for which T=1-P. The values of n_0, n_1 which maximize throughput for S=1000 blocks (q=1) are shown in Table (3.1) . It might be noted that the improvement in T should decrease if more buffering capabilities are available in the receiver for the above example. This is shown in Figure (3.2) and Figure (3.3) in which the throughput T is plotted against P for S=1000 and q=2, q=3 respectively. The reason why the improvement in T decreases can be explained by the fact that the probability of receiver buffer overflow decreases with increasing q. Table 3.2 shows the values of n_0,n_1 ,n_2,and n_3 which maximize T for S=1000 blocks and q=3.

TABLE 3.1

VALUES OF n_0 AND n_1 WHICH MAXIMIZE THROUGHPUT FOR S=1000BLOCKS WITH q=1 FOR SEVERAL VALUES OF P

P	n_o	n_1	T
0.01	1	2	0.97943
0.04	1	3	0.89082
0.10	1	3	0.71413
0.20	1	5	0.48443
0.30	2	6	0.38373
0.40	2	7	0.29550
0.50	3	9	0.22875
0.60	4	12	0.17120
0.70	5	17	0.12113
0.80	8	24	0.07788
0.90	17	45	0.03841

TABLE 3.2

VALUES OF n_0, n_1, n_2, AND n_3 WHICH MAXIMIZE THROUGHPUT FOR S=1000BLOCKS WITH q=3 FOR SEVERAL VALUES OF P

P	n_0	n_1	n_2	n_3	T
0.1	1	1	1	3	0.89767
0.2	1	1	1	5	0.77648
0.3	1	1	2	6	0.64517
0.4	1	1	2	8	0.50471
0.5	1	2	3	10	0.38240
0.6	2	2	4	13	0.27739
0.7	2	3	6	17	0.19364
0.8	4	5	9	26	0.11713
0.9	7	10	16	45	0.06232

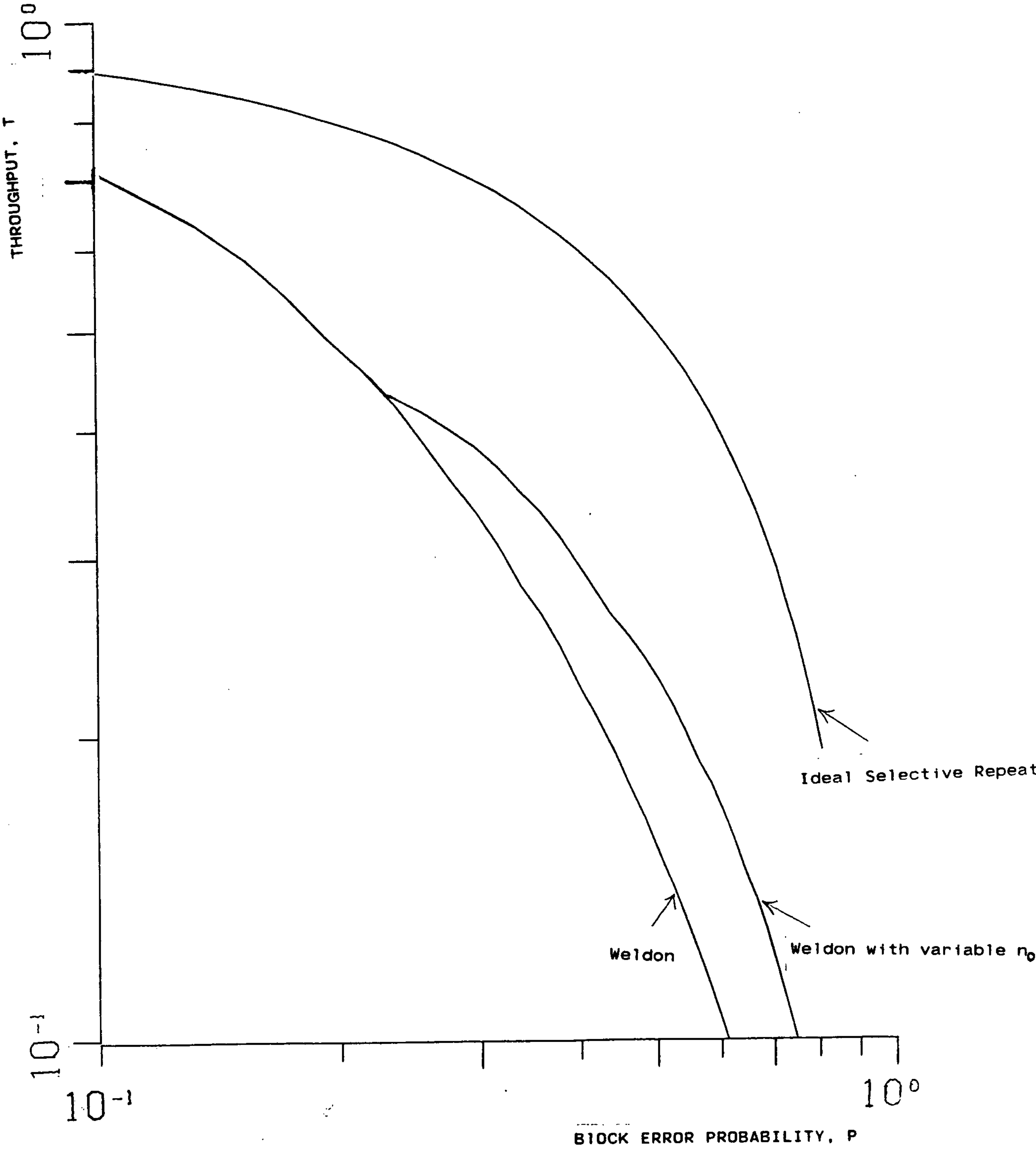

Fig 3.1-Plot of throughput against block error probability with optimum values of $\{n_i\}$ for q=1 and S=1000 blocks.

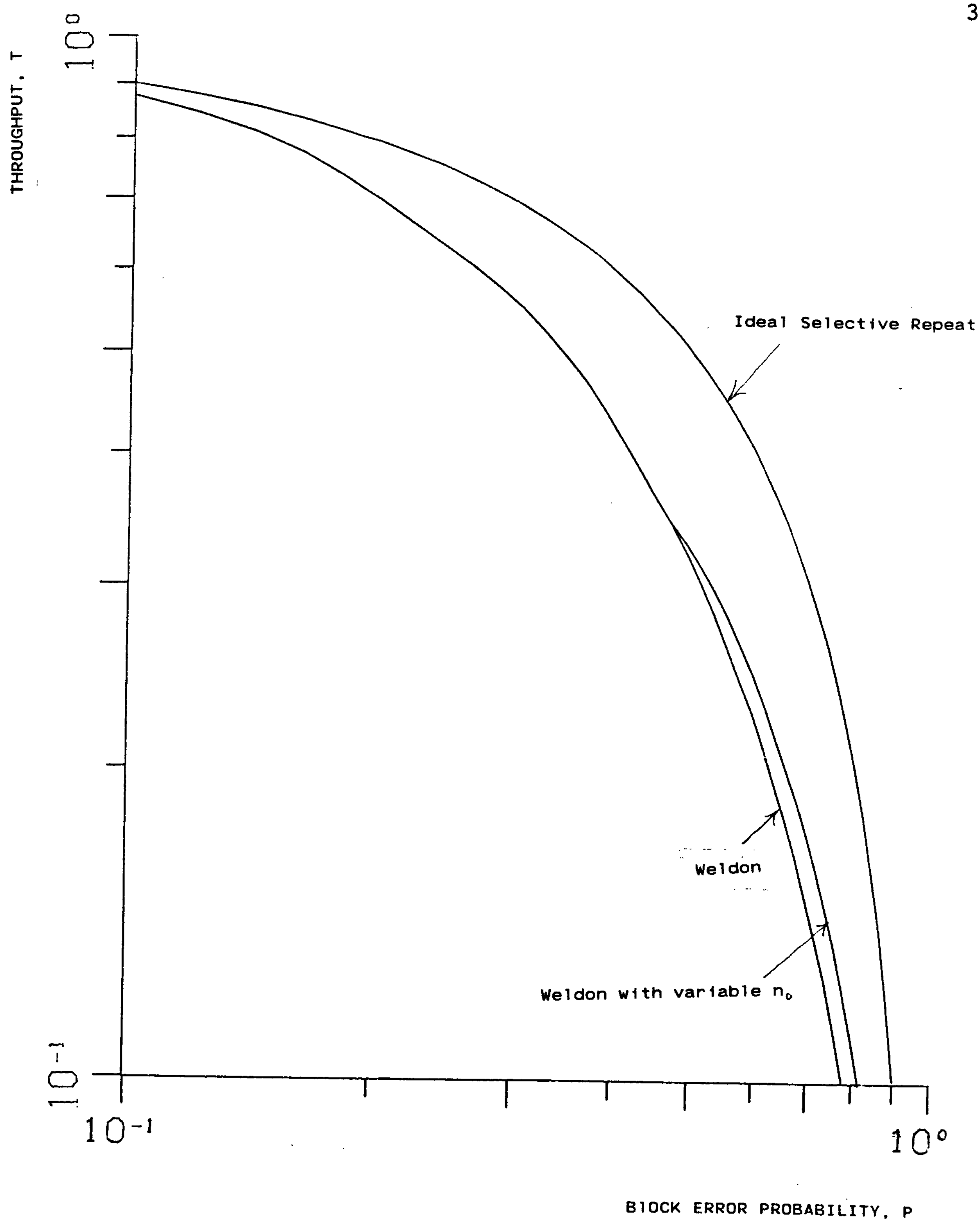

Fig 3.2-Plot of throughput against block error probability with optimum values of $\{n_i\}$ for q=2 and S=1000 blocks.

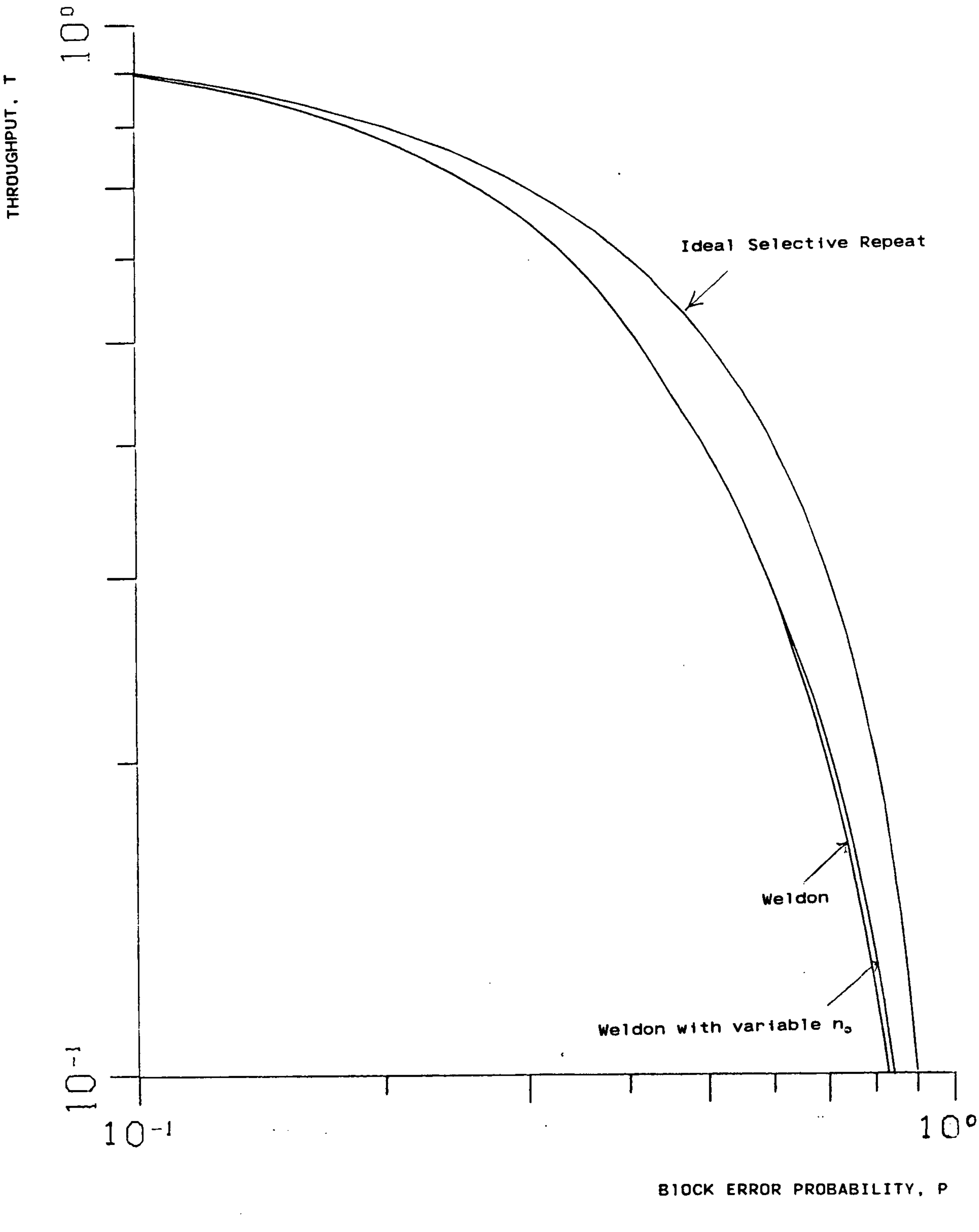

Fig 3.3-Plot of throughput against block error probability with optimum values of $\{n_i\}$ for q=3 and S=1000 blocks.

3.4 Determination Of The Optimum Values Of $\{n_i\}$

In order to maximize the throughput, a method must be devised to chose the optimum values of the parameters $\{n_i\}_{i=0}^{q}$. One obvious method is to use brute force searching, For example, the optimum values of n_0 and n_1 for the cases of $q=1$ can be obtained as follows: (The convexity of $\beta(n_0,n_1)$ is shown in Appendix B)

step 0: Initialize $n_0=1$.

step 1: Compute $\beta(n_0,n_1)$ of equation (3.2.2) for $n_1=1,2,3,4,,\ldots$. until the value of n_1^* which minimizes β is found.

step 2: Increment n_0 by 1 .

step 3: Go back to step 1 until the pair of values (n_0^*,n_1^*) that minimizes β is found.

For the cases of $q=1$, the above method takes $\cong n_0^* \cdot n_1^*$ searches to find the optimum pair of values (n_0^*,n_1^*) . In general, the brute force searching method takes $\cong n_0^* n_1^* n_2^* n_3^* \ldots n_q^*$ searches to find the optimum values of $\{n_i\}_{i=0}^{q}$. However, a more efficient method for choosing these parameters can be derived by exploiting the structure of the simplified throughput expression ,equation (3.2.2). We first note that equation (3.2.2) can be re-written as

$$\beta = n_0 + n_1 P^{n_0} + n_2 P^{n_0+n_1} + \ldots + n_i P^{n_0+n_1+\ldots+n_{i-1}} + \ldots$$

$$+ n_q P^{n_0+n_1+\ldots+n_{q-1}} + \frac{(n_q+S-1)P^{n_0+n_1+\ldots+n_q}}{1-P^{n_q}}$$

$$= n_0 + P^{n_0}(n_1 + P^{n_1}(n_2 + P^{n_2}(\ldots(n_i + P^{n_i}(\ldots$$

$$(n_{q-1} + P^{n_{q-1}}(n_q + P^{n_q}(\frac{n_q+S-1}{1-P^{n_q}})))\ldots))\ldots)))) \tag{3.4.1}$$

Equation (3.4.1) shows that the minimization of β can be performed by using the following procedure:

step 0 : Determine the values of n_q (say n_q^*) which minimizes

$$f_q(n_q) = n_q + P^{n_q}\left(\frac{n_q + S - 1}{1 - P^{n_q}}\right) \qquad (3.4.2)$$

step 1 : Determine the value of n_{q-1} (say n_{q-1}^*) which minimizes

$$f_{q-1}(n_{q-1}) = n_{q-1} + P^{n_{q-1}} f_q(n_q^*)$$

$$\vdots \qquad (3.4.3)$$

step q-i. Determine the value of n_i (say n_i^*) which minimizes

$$f_i(n_i) = n_i + P^{n_i}\, f_{i+1}(n_{i+1}^*, n_{i+2}^*, \ldots, n_q^*) \qquad (3.4.4)$$

$\vdots$

step q . Determine the value of n_0 (say n_0^*) which minimizes

$$f_0(n_0) = n_0 + P^{n_0} f_1(n_1^*, n_2^*, \ldots, n_q^*) \qquad (3.4.5)$$

The above procedure indicates that the optimum value of n_i depends only on S, P and the optimum values of $n_{i+1}, n_{i+2}, \ldots n_q$. As an example of how the procedure can be used to derive the optimum values of $\{n_i\}$ we consider the case when q=1. It is shown in Appendix B that the functions $\{f_i(n_i)\}_{i=0}^{q}$ are all convex functions. Hence,the necessary and sufficient conditions for n_1^* to be the optimum value for n_1 are

$$f_1(n_1^*) < f_1(n_1^* - 1) \tag{3.4.6}$$

and

$$f_1(n_1^*) < f_1(n_1^* + 1). \tag{3.4.7}$$

Using $f_1(n_1)$ as given in equation (3.4.2), it is readily shown that n_1^* is the optimum value if and only if

$$\frac{P^{-(n_1^* - 1)} - 1}{(1 - P)} - n_1^* + 2 < S < \frac{P^{-n_1^*} - 1}{(1 - P)} - n_1^* + 1. \tag{3.4.8}$$

Weldon [6] gives the following guidelines for choosing n_1 :

$$\begin{aligned} &n_1 = 1, \ 0 < SP < 1 \\ &n_1 = 2, \ 1 < SP \text{ and } 0 < SP^2 < 1 \\ &n_1 > 3, \ 1 < SP^2. \end{aligned} \tag{3.4.9}$$

From equation (3.4.8), the exact rules for choosing n_1 are:

$$\begin{aligned} &n_1^* = 1, \ SP < 1. \\ &n_1^* = 2, \ 1 < SP \text{ and } SP^2 < 1 + P - P^2 \\ &n_1^* = i, \ (1+P+ \ldots +P^{i-2}) - (i-2)P^{i-1} < SP^{i-1} \text{ and } SP^i < \\ &\qquad (1+P+ \ldots +P^{i-1}) - (i-1)P^i. \end{aligned} \tag{3.4.10}$$

As noted earlier, in [6] it is assumed that n_0 is 1. Recalling from equation (3.4.4) with q=1 that

$$f_0(n_0) = n_0 + P^{n_0} f_1(n_1^*) \tag{3.4.11}$$

we obtain the optimum value of n_0 by using

$$f_0(n_0^*) < f_0(n_0^* - 1)$$

and

$$f_0(n_0^*) < f_0(n_0^* + 1).$$

(3.4.12)

Figure (3.4) shows a plot of the values of P and S for which a given value of n_0 is optimum. For example, if P=0.3 and S=1000, the optimum value for n_0 is 2. A similar plot for n_1 is given in Figure (3.5) . It can be seen that for P=0.3 and S=1000,the optimum value for n_1 is 6.

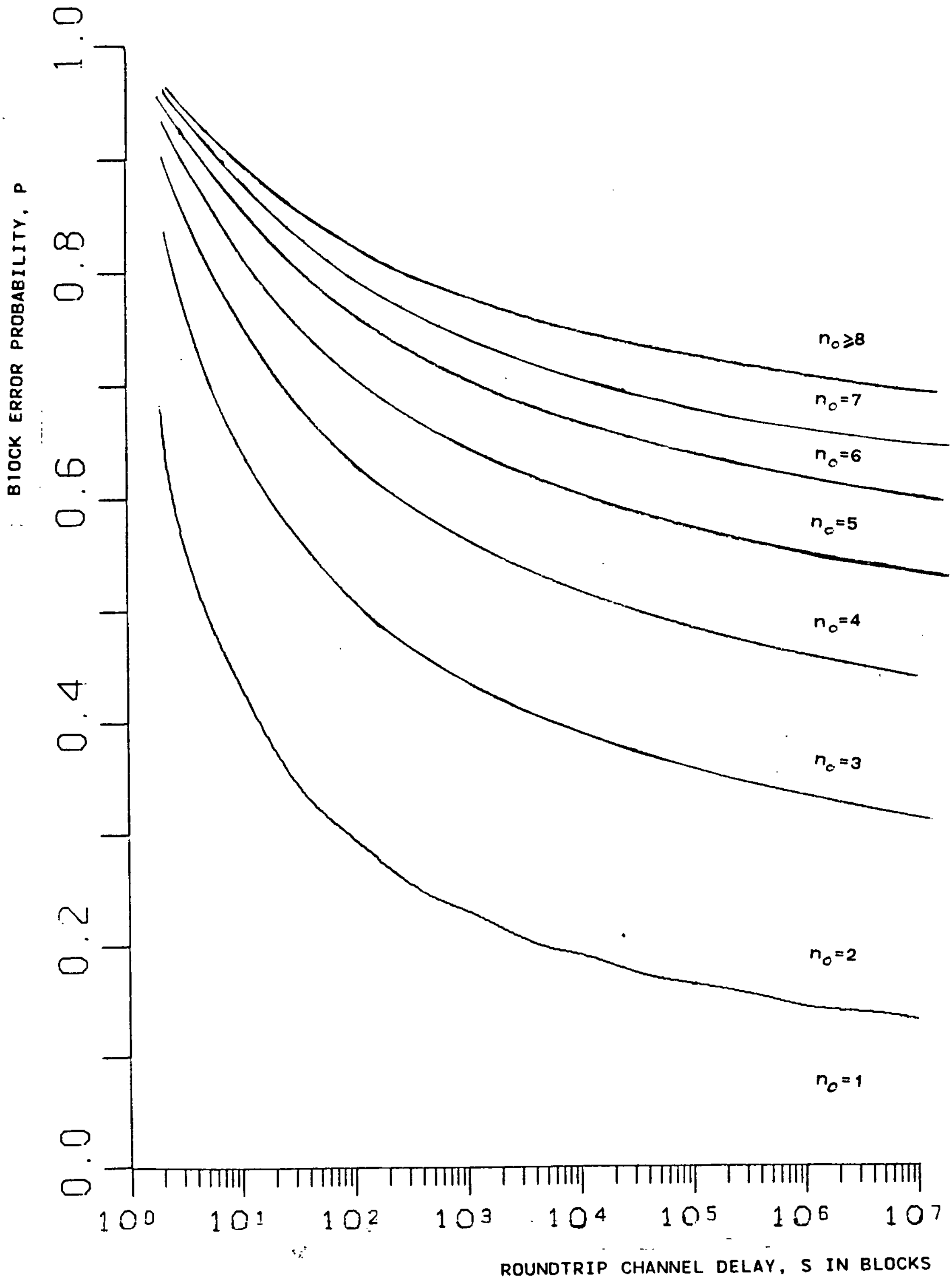

Fig 3.4-Optimum value of n_o for the Weldon with variable n_o scheme with q=1.

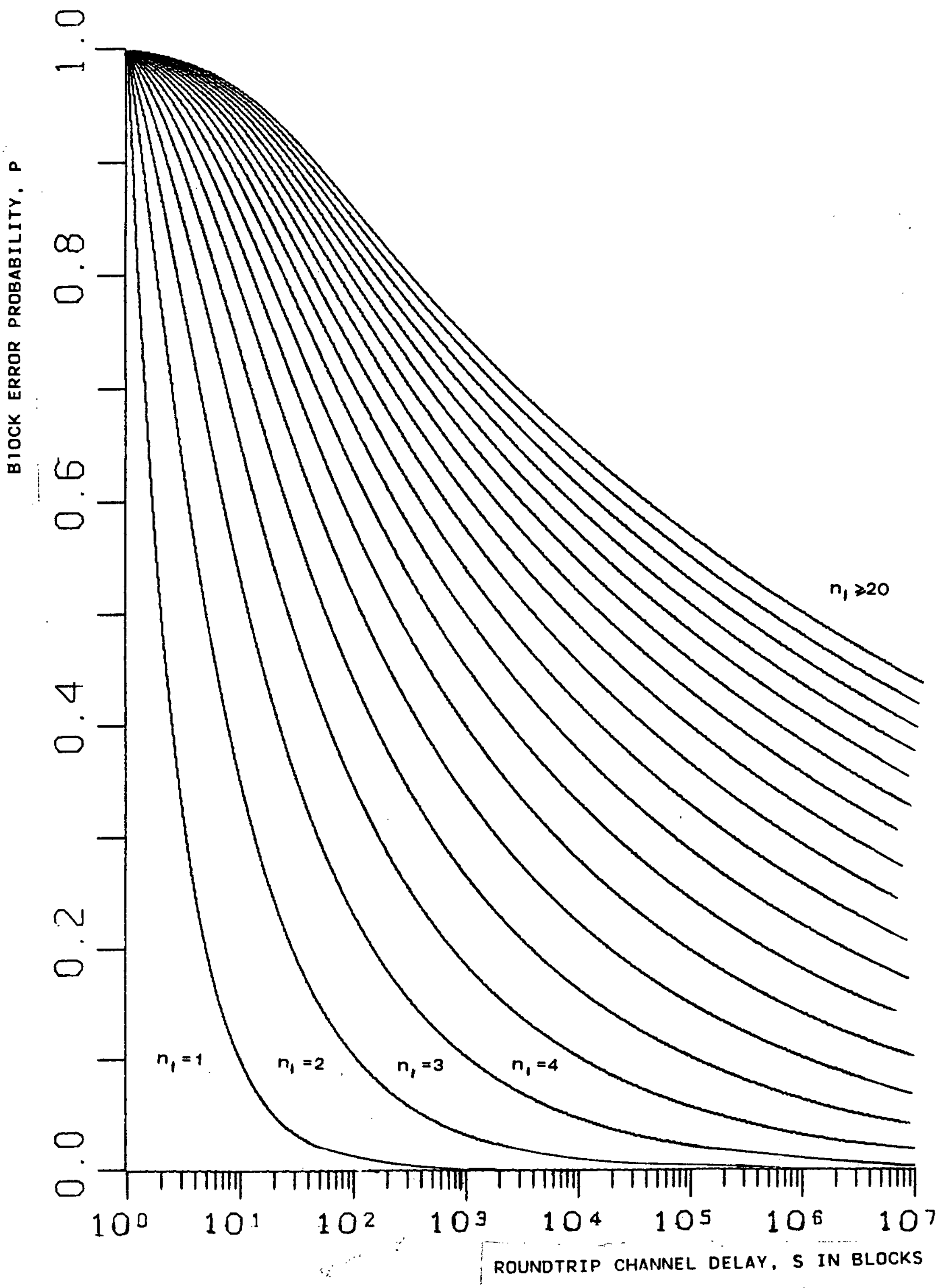

Fig 3.5-Optimum value of n_1 for the Weldon with variable n_o scheme with q=1.

3.5 A Modified Weldon ARQ Scheme

It has been shown that the throughput of Weldon's scheme can be improved by allowing multiple copies of a new block to be transmitted. Another modification is to prevent buffer overflow (at level q+1) by repeating any block which is at level q continuously until an ACK is received.The analysis of this modified scheme leads to the following expression for β :

$$\beta = \sum_{i=0}^{q-1} \left(\sum_{j=0}^{i} n_j\right) P^{\left(\sum_{j=0}^{i-1} n_j\right)} \left(1-P^{n_i}\right)$$

$$+ \sum_{i=0}^{\infty} \left(i+(S-1)+\sum_{j=0}^{q-1} n_j\right)(1-P) P^{\left(\sum_{j=0}^{q-1} n_j + i - 1\right)} \tag{3.5.1}$$

After simplification , equation (3.5.1) is reduced to

$$\beta = \sum_{i=0}^{q-1} n_i \, P^{\sum_{k=0}^{i-1} n_k} + \left(S-1+\frac{1}{1-P}\right) P^{\sum_{j=0}^{q-1} n_j} . \tag{3.5.2}$$

It might be noted that Morris' GBN scheme in [4] is a special case of this modified scheme for q=1, and $n_0 = 1$. Also, Miller and Lin's SR+ST I [5] is a special case of this scheme with n_i set to 1 for all i.

Unfortunately , it was found that this modified scheme yields a higher throughput than the Weldon with variable n_0 scheme only for high values of P. For example, with S=10, the threshold value for P is 0.88 and with S=50, the threshold value is 0.98.

It might be concluded that this modified Weldon scheme should be considered for use only for channels with very high P.

IV. OPTIMAL BLOCK LENGTH ANALYSIS

4.1 An Exact Analysis Of Optimal Block Length

As illustrated in previous chapters,the expected wasted time performances of all ARQ schemes critically depend on the choice of the packet length. For given values of $R, A, P_b, \overline{L}$ and b ,an optimum block length that minimizes the expected wasted time should exist [9]. Morris[10] gave an analysis of optimum block length for some standard ARQ schemes. However, Morris' analysis neglects the overhead b for addressing, error control, etc. in each packet, which is an important parameter to be considered. Chu [9] gave the solutions of optimum packet length in the form of non-explict equation for the Stop-And-Wait,Go-Back-N, and Ideal Selective-Repeat schemes. However, Chu's analysis of the optimal packet length for the Stop-And-Wait scheme using an end-of-message character is inexact (Appendix C shows Chu's analysis[9]). In this section, an exact analysis of the expected wasted time per message is given for the case of using end-of-message character for the Stop-And-Wait scheme. It is found that the optimum block length derived from the exact analysis can lead to some improvement of expected wasted time for small values of $\overline{L}$. The analysis is shown as follows.

We assume the messages have random message lengths L which are geometrically distributed with average length $\overline{L}=1/(1-q)$. Each message is split into blocks of size B bits. These blocks are then assembled into packets of length (B+b) bits. To increase channel efficiency, an end-of-message character is used to designate the end of the last unfilled block rather than fill the rest of the block with dummy information .Thus, the length of the last packet is between (1+b) bits and (B+b) bits.

For the Stop-And-Wait scheme, the exact expression for the expected wasted time per message incurred over a random-error channel is:

$$\overline{W}_e(B)=\sum_{\ell=1}^{\infty}(1-q)q^{\ell-1}\Big\{\Big\lfloor\frac{\ell}{B}\Big\rfloor\Big(\frac{B+b}{R}+A\Big)\frac{1}{(1-P_b)^{B+b}} + F\cdot\Big(\frac{\ell-\lfloor\frac{\ell}{B}\rfloor B+b}{R}+A\Big)\frac{1}{(1-P_b)^{\ell-\lfloor\frac{\ell}{B}\rfloor B+b}} - \frac{\ell+b}{R}\Big\} \tag{4.1.1}$$

where

$$F(\ell)=\begin{cases}1 & \text{if } \ell/B \text{ is not an integer.}\\ 0 & \text{if } \ell/B \text{ is an integer.}\end{cases}$$

and $\lfloor\ \rfloor$ is the floor function.

Equation 4.1.1 can be rewritten as:

$$\begin{aligned}\overline{W}_e(B)=&\sum_{\ell=1}^{B-1}(1-q)q^{\ell-1}\Big(\frac{\ell+b}{R}+A\Big)\frac{1}{(1-P_b)^{\ell+b}}-\sum_{\ell=1}^{\infty}(1-q)q^{\ell-1}\Big(\frac{\ell+b}{R}\Big)\\ &+(1-q)q^{B-1}\Big(\frac{B+b}{R}+A\Big)\frac{1}{(1-P_b)^{B+b}}+\sum_{\ell=B+1}^{2B-1}(1-q)q^{\ell-1}\Big(\frac{B+b}{R}+A\Big)\frac{1}{(1-P_b)^{B+b}}\\ &+\sum_{\ell=B+1}^{2B-1}(1-q)q^{\ell-1}\Big(\frac{\ell-B+b}{R}+A\Big)\frac{1}{(1-P_b)^{\ell-B+b}}\\ &+2(1-q)q^{2B-1}\Big(\frac{B+b}{R}+A\Big)\frac{1}{(1-P_b)^{B+b}}+\sum_{\ell=2B+1}^{3B-1}(1-q)q^{\ell-1}\Big(\frac{\ell-2B+b}{R}+A\Big)\frac{1}{(1-P_b)^{\ell-2B+b}}\\ &+\sum_{\ell=2B+1}^{3B-1}(1-q)q^{\ell-1}2\Big(\frac{B+b}{R}+A\Big)\frac{1}{(1-P_b)^{B+b}}\\ &\qquad\vdots\qquad\qquad\vdots\end{aligned} \tag{4.1.2}$$

By a change of variables in the summation terms and making use of the equalities $\sum_{i=0}^{\infty} q^i = 1/(1-q)$ and $\sum_{i=0}^{\infty} i q^{i-1} = 1/(1-q)^2$, equation (4.1.2) can be simplified to

$$\overline{W}_e(B) = \frac{1-q}{(1-q^B)} \cdot \frac{1}{(1-P_b)^{b+1}} \left\{ \frac{1-B\left(\frac{q}{1-P_b}\right)^{B-1} + (B-1)\left(\frac{1}{1-P_b}\right)^B}{R\left(1-\frac{q}{1-P_b}\right)^2} + \left(\frac{b}{R}+A\right) \cdot \frac{1-\left(\frac{q}{1-P_b}\right)^{B-1}}{1-\frac{q}{1-P_b}} \right\}$$
$$+ \left(\frac{B+b}{R}+A\right) \frac{q^B}{(1-P_b)^{B+b}(1-q^B)^2} (1-q^{B-1}+(1-q)q^{-1}) - \frac{\overline{L}+b}{R} \tag{4.1.3}$$

Therefore, the block length B' that minimizes the expected wasted time for the Stop-And-Wait scheme using an end-of-message character can be obtained from searching for the B',such that $\overline{W}_e(B')$ is a global minimum. In Figure (4.1), the optimum block lengths B' obtained from equation (4.1.3) is plotted against the average message length $\overline{L}$. Also plotted in Fig (4.1) are the optimum block lengths B_c obtained from Chu's inexact expression (equation (C.5)) for the case of using an end-of-message character and the optimum block length for the case of using dummy bits(equation (2.2.4)). From Fig (4.1) , it can be seen that the block lengths B_c differ significantly from B' except for large values of average message length. Table (4.1) shows the percentage differences in expected wasted times which result when B' and B_c are used in equation (4.1.3) . The results show that for small $\overline{L}$,say $\overline{L}<200$, some improvement in expected wasted time can be achieved by using B' instead of B_c.

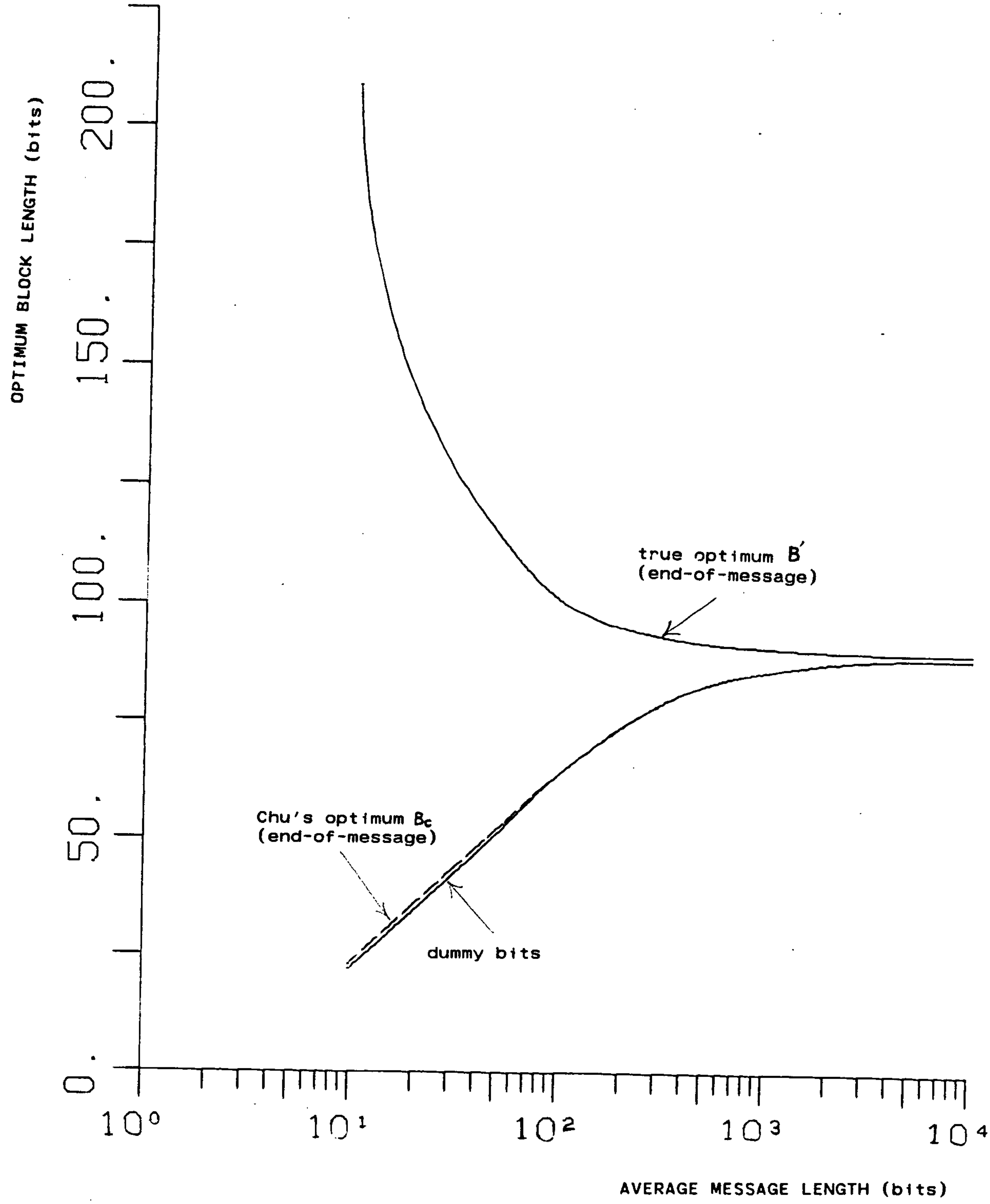

Fig 4.1-Plot of optimum block length against average message length for Stop-And-Wait scheme with P_b=0.01,R=4000bits/s,A=0.2s, and b=30 bits.

TABLE 4.1

PERCENTAGE DIFFERENCE IN EXPECTED WASTED TIME FOR

R=4000BITS/SEC,A=0.2SEC,b=30BITS,P_b=0.01

			EXPECTED WASTED TIME $\overline{W}_e$ (equation (4.1.3))		
$\overline{L}$	B'	B_c	$\overline{W}_e(B')$	$\overline{W}_e(B_c)$	$(\overline{W}_e(B_c)-\overline{W}_e(B'))/\overline{W}_e(B')\times100\%$
10	209	23	0.305	0.332	7.9%
50	115	52	0.542	0.614	13.2%
100	101	64	0.937	1.001	6.8%
500	92	83	4.238	4.258	0.5%
1000	91	86	8.380	8.392	0.1%

4.2 Asymptotic Values Of Optimum Block Length

It should be noted that computing the optimum block lengths from equations (4.1.3),(C.4), or (2.2.4) requires the knowledge of average message length $\overline{L}$. In practice, it may not be easy or convenient to have an accurate measurement of $\overline{L}$. In Fig (4.1), it appears that there is an asymptotic value for the optimum block length when $\overline{L}$ becomes large. One way of avoiding the need to know $\overline{L}$ is to use the asymptotic value rather than $B'(\overline{L})$. In this section, a method to compute the asymptotic value of the optimum block length is given . The performance of using this asymptotic value is also examined for the standard Stop-And-Wait and Go-Back-N schemes.

We proceed by introducing an approximate expression for the wasted time to successfully transmit a packetized message of length ℓ:

$$W_A \cong \frac{\ell}{B} \cdot \frac{1}{T} \cdot D - \frac{\ell+b}{R} \qquad (4.2.1)$$

where T=throughput of the ARQ scheme used.

D=(B+b)/R=time to a packet in sec.

Obviously, the above equation is close to exact if ℓ is very large compared to B. By differentiating equation (4.2.1) with respect to B and setting the derivative to zero,

$$\frac{\partial W_A}{\partial B} = \frac{\partial}{\partial B}\left(\frac{D}{B \cdot T}\right) = 0 \qquad (4.2.2)$$

the asymptotic value B^∞ of optimum block length can be obtained. For the Stop-And-Wait scheme(assuming random-error channel) with T=(1-P)/S (as given in equation (2.1.1)), B^∞ is :

$$B^\infty = \frac{1}{2}(RA+b)\left(\sqrt{1 - \frac{4}{(RA+b)\ln(1-P_b)}} - 1\right) \qquad (4.2.3)$$

where A=(S-1)D=acknowledgment delay in sec.

The derivation of equation (4.2.3) is shown in Appendix E. Similarly, for the Go-Back-N scheme with T=(1-P)/(1+(S-1)P) (as given in equation (2.1.2)), B^∞ is obtained by solving :

$$B\left(\frac{B+b}{R}+A\right)\ln(1-P_b) - A(1-P_b)^{B+b} + \left(A+\frac{b}{R}\right) = 0 \qquad (4.2.4)$$

Table (4.2) shows the B^∞ solved from equation (4.2.3) for several values of P_b . It is seen that for P_b=0.01, B^∞=90 bits which is the same as the asymptotic value of optimum block length shown in Fig (4.1) .

TABLE 4.2

ASYMPTOTIC VALUES OF OPTIMUM BLOCK LENGTH FOR STOP-AND-WAIT SCHEME WITH R=4000 BITS/SEC, A=0.2SEC, b=30 BITS

P_b	B^{∞}
0.1	9
0.05	19
0.01	90
0.005	166
0.001	586
0.0005	938
0.0001	2496

Also, Table (4.3) shows the B^{∞} solved from equation (4.2.4) for several values of P_b .

TABLE 4.3

ASYMPTOTIC VALUES OF OPTIMUM BLOCK LENGTH FOR GO-BACK-N SCHEME FOR R=4000 BITS/SEC, A=0.2SEC, b=30 BITS

P_b	B^{∞}
0.1	9
0.05	17
0.01	55
0.005	82
0.001	179
0.0005	248

We now examine the performance degradation in expected wasted time which results if B^{∞} is used instead of $B'(\overline{L})$. Tables (4.4), (4.5) and (4.6) show the percentage differences in expected wasted time between using B^{∞} and using B' for the Stop-And-Wait scheme(end-of-message character and dummy bits cases) and Go-Back-N scheme (dummy bits case) for several values of $\overline{L}$.

TABLE 4.4

PERCENTAGE DIFFERENCE IN EXPECTED WASTED TIME BETWEEN USING B^{∞} AND B' FOR THE STOP-AND-WAIT SCHEME (END-OF-MESSAGE CHARACTER CASE) WITH B^{∞}=90BITS, R=4000 BITS/SEC, A=0.2SEC, P_b=0.01, b=30 BITS.

$\overline{L}$	B'	$W_e(B^{\infty})$	$W_e(B')$	$(W_e(B^{\infty})-W_e(B'))/W_e(B')\times100\%$
10	209	0.305	0.305	0.0%
50	115	0.548	0.542	1.0%
100	101	0.940	0.937	0.4%
500	92	4.239	4.238	0.0%
1000	91	8.379	8.379	0.0%

TABLE 4.5

PERCENTAGE DIFFERENCE IN EXPECTED WASTED TIME BETWEEN USING B^{∞} AND B' FOR THE STOP-AND-WAIT SCHEME (DUMMY BITS CASE) WITH B^{∞}=90BITS,R=4000BITS/SEC, A=0.2SEC,P_b=0.01 AND b=30BITS.

$\overline{L}$	B'	$W(B^{\infty})$	W(B')	$(W(B^{\infty})-W(B'))/W(B')\times100\%$
10	22	0.758	0.388	95.1%
50	51	0.897	0.731	22.7%
100	64	1.258	1.179	6.6%
500	83	4.526	4.510	0.3%
1000	86	8.664	8.655	0.1%

TABLE 4.6

PERCENTAGE DIFFERENCE IN EXPECTED WASTED TIME BETWEEN USING B^{∞} AND B' FOR THE GO-BACK-N SCHEME (DUMMY BITS CASE) WITH B^{∞}=55BITS,R=4000 BITS/SEC A=0.2SEC,P_b=0.01,b=30BITS.

$\overline{L}$	B'	$W(B^{\infty})$	W(B')	$(W(B^{\infty})-W(B'))/W(B')\times100\%$
10	15	0.311	0.157	98.0%
50	32	0.457	0.404	13.1%
100	39	0.721	0.691	4.3%
500	51	2.935	2.929	0.2%
1000	53	5.717	5.715	0.0%

From the example in Table (4.4), it is seen that the use of B^{∞} makes little difference in the expected wasted time for the Stop-And-Wait scheme using an end-of-message character. Thus, one might suspect that the expected wasted time for the Stop-And-Wait scheme using end-of-message character is not very sensitive to the block length used as long as $B \gg \bar{L}$ for small $\bar{L}$. For the Go-Back-N and Stop-And-Wait schemes (using dummy bits), there are significant percentage differences in expected wasted time for small values of $\bar{L}$ (i.e. $\bar{L}$<100bits) , as shown in Tables (4.5) and (4.6). However, using B^{∞} gives almost the same expected wasted time as using B' for larger values of $\bar{L}$(i.e. $\bar{L}$>100bits) in all three cases.

4.3 Adaptive Algorithm For Optimal Block Length Computation

For the optimal block length analysis cited above, the block length is optimized to give the minimum expected wasted time per message for messages of average length $\bar{L}$. However,the block length that minimizes the expected wasted time per message for messages of average length $\bar{L}$ does not necessarily give the minimum wasted time for each individual message. Therefore, one could consider computing the optimal block length for each individual message of length ℓ. In this section, such an optimal block length strategy is presented with an algorithm that computes the optimal block length in an adaptive way for the Stop-And-Wait scheme. The idea of the algorithm is to search for the optimum combination of number of packets(denoted by N) and block lengths for transmitting a message of length ℓ. It might be noted that executing the algorithm requires a knowledge of the message length. This information can be obtained by scanning the message while it is stored in the buffer queue waiting for packetizing .For a message of length ℓ, the adaptive algorithm for packetizing is given as follows :

step 1: set N=0

step 2: Increment N by 1.

step 3: Compute the block length $B=\lfloor \ell/N \rfloor$ and the remaining bits $M=\ell-N\cdot B$

step 4: Distribute the M bits into the N packets ,resulting in M packets of length (B+1) bits and (N-M) packets of length B bits.

step 5: Compute the wasted time of transmitting these N packets for the Stop-And-Wait scheme(for random-error channel) without using dummy bits.

$$W_s(\ell)= M\cdot\left(A+\frac{B+1+b}{R}\right)\frac{1}{(1-P_b)^{B+1+b}}+(N-M)\cdot\left(A+\frac{B+b}{R}\right)\frac{1}{(1-P_b)^{B+b}}-\frac{\ell+b}{R} \tag{4.3.1}$$

step 6: Go back to step 2 until the value of N that gives the minimum wasted time $W_S(\ell)$ is found.

Although it is not possible to prove the convexity of W_S , all the simulation results indicate W_S is a convex function of N.

Notice in step 4, the remaining M bits are distributed into the N packets so that the lengths of the N packets are equalized into B bits or (B+1) bits. In Appendix D, it is shown that such an equalization of packet length always gives better or equal performance in the wasted time of transmitting a packetized message.

The performance of this algorithm is examined with geometric message length distribution and random error channel. The expected wasted time $W_S(\bar{L})$ per message ($\bar{L}=1/(1-q)$) for the system using the adaptive optimal block length algorithm is:

$$\bar{W}_s(\bar{L}) \gtrsim \sum_{\ell=1}^{max} (1-q)q^{\ell-1}.W_s(\ell) \qquad (4.3.2)$$

where $W_S(\ell)$ is the minimum wasted time computed from the adaptive algorithm for a message of length ℓ.

Table (4.7) shows the expected wasted time $\bar{W}_s(\bar{L})$ computed from equation (4.3.2) with max set to a sufficient large number , say 10000 , along with $\bar{W}_e(\bar{L})$ computed from equation (4.1.3) using the conventional optimal block length approach. It might be noted that $\bar{W}_s(\bar{L})$ should increase with max. However, for this example, the percentage increase in $\bar{W}_s(\bar{L})$ is on the order of 10^{-5} if max is increased from 10000 to 50000.

TABLE 4.7

PERCENTAGE IMPROVEMENT OF EXPECTED WASTED TIME FOR THE ADAPTIVE ALGORITHM FOR OPTIMAL BLOCK LENGTH COMPUTATION WITH R=4000BITS/SEC, A=0.2SEC,P_b=0.01 AND b=30BITS.

$\bar{L}$	$\bar{W}_s(\bar{L})$	$\bar{W}_e(\bar{L})$	$(\bar{W}_e-\bar{W}_s)/\bar{W}_s$ x100%
10	0.305	0.305	0.0%
50	0.529	0.548	3.6%
100	0.900	0.940	4.3%
500	4.160	4.238	1.8%
1000	8.289	8.379	1.1%

The results in Table (4.7) show that the adaptive algorithm yields some improvement in expected wasted time performances for some values of $\overline{L}$. For small values of $\overline{L}$ (i.e. $\overline{L}<50$), the adaptive algorithm is not effective since a message is most likely to be transmitted as one block. Also, when $\overline{L}$ becomes large (i.e. $\overline{L}>500$), the improvement decreases since the effect of the last packet on the overall expected wasted time is less significant. It is found that the percentage improvement is not very sensitive to R ,A or b. However, as the bit error rate P_b gets smaller, less improvement is expected. It should be noted that a system implementing such an adaptive algorithm has the extra costs for scanning the length of the messages and executing the algorithm .

V. CONCLUSIONS

5.1 SUMMARY OF RESULTS

In this thesis , the performances of a number of ARQ schemes are compared on the basis of expected wasted time per message. The reductions in expected wasted times achievable through the used of FEC are also demonstrated. It is interesting to note that FEC tends to equalize the expected wasted times of the continuous ARQ schemes.

A new ARQ scheme proposed by Weldon [6] is investigated. It is found that the throughput of Weldon's scheme can be substantially increased by allowing multiple copies of a new data block to be sent. An efficient method is also developed for determining the parameters of Weldon's scheme which maximize throughput.

An exact analysis of the optimal block length is given for the Stop-And-Wait scheme using an end-of-message character.It is found that there is an asymptotic value for the optimal block length as the average message length becomes large. An algorithm that computes the optimal block length on an individual message basis is also developed .

5.2 Suggestions For Future Research

The analysis in this thesis is restricted to point-to-point channels. Attempts could be made to extend this analysis to broadcast channels .Mase,Takenaka and Yamamoto [19] gave an analysis for the Go-Back-N protocol in a broadcast environment. Similar studies could be made for other ARQ schemes.i.e. SR+ST

, Weldon ,etc.It might be noted that if these schemes are to be used in a broadcast environment, some additional logic may need to be implemented in the transmitter and receiver so as to be applicable to point-to-multipoint channels.

APPENDIX A - DESCRIPTION OF WELDON'S ARQ SCHEME

Weldon's ARQ scheme and its throughtput analysis are briefly described here. The full description can be found in [6].

In Weldon's scheme, the receiver buffer is of size qS where q is a positive integer. The transmission state of each block is described by its level. Each block is transmitted according to the following procedure:

level 0: Initially, B is at level 0 and is transmitted for the first time. If an ACK is received (S block times later) the transmission of B is complete. If an NACK is is received, B moves to level 1.

level 1: B is repeated n_1 times.If any of these n_1 copies is ACKed,the transmission of B is complete. If all n_1 copies are NACKed, B moves to level 2.

level 2: B is repeated n_2 times. If any of these n_2 copies is ACKed, the transmission of B is complete. If all n_2 copies are NACKed, B moves to level 3.

.
.
.

level q: If all n_{q-1} copies are in error, the receiver buffer is considered full even though it may not actually be full because of repeats of other erroneoue blocks. This assumption leads to a simple analysis of throughput If all n_q copies are NACKed , B moves to level 3.

level q+1:Buffer overflow occurs leading to the loss of (S-1) blocks. B is repeated n_q times and stays at this level until it is sucessfully received.

It is understood that whenever the transmitter does not have any repeats to send, it transmits new blocks of data.Equation (8) of [6] shows that the average number of transmissions require to successfully send one block =1/T is

$$\beta = \sum_{i=0}^{q} \left(\sum_{j=0}^{i} n_j\right) P^{\left(\sum_{j=0}^{i-1} n_j\right)} (1-P^{n_j})$$
$$+ \sum_{k=1}^{\infty} \left(k(n_q+S-1) + \sum_{j=0}^{q} n_j\right) \cdot (1-P^{n_q}) P^{\left(\sum_{j=0}^{q} n_j + (k-1)n_q\right)} \quad (A.1)$$

APPENDIX B - PROOF OF CONVEXITY

Lemma: The functions $\{f_i(x)\}_{i=0}^{q}$, $x > 0$, defined by equation (3.4.2-3.4.5) are convex.

Proof: We proceed by first proving that $f_q(x)$ is convex. It can then be easily shown that $\{f_i(x)\}_{i=0}^{q-1}$ are also convex.

From (3.4.2) we can write

$$f_q(x) = x + P^x\left(\frac{x+c}{1-P^x}\right) \text{ where } c \triangleq S-1. \tag{B.1}$$

Taking the derivative of (B.1) twice, we obtain

$$f_q''(x) = (1-P^x)^{-3}P^x(\ln P)[2(1-P^x) + (x+c)(1+P^x)\ln P] \tag{B.2}$$

Since $\ln P < 0$, $f_q''(x) > 0$ (i.e. $f_q(x)$ will be convex) if and only if

$$2(1-P^x) + (x+c)(1+P^x)\ln P < 0. \tag{B.3}$$

Since $c>0$, a sufficient condition for $f_q''(x) > 0$ is that

$$g(x) \triangleq x(1+P^x)\ln\frac{1}{P} - 2(1-P^x) > 0. \tag{B.4}$$

Now,

$$g'(x) = (\ln\frac{1}{P})\, h(x) \tag{B.5}$$

where

$$h(x) \triangleq 1 - P^x(1+x\ln\frac{1}{P}).$$

Also,

$$h'(x) = xP^x(\ln \frac{1}{P})^2. \tag{B.6}$$

Since $h'(x) > 0$ for $x > 0$ and $h(0) = 0$, we have that $h(x) > 0$ for $x > 0$. From (3.4.3) it follows that $g'(x) > 0$ for $x > 0$.

But $g(0) = 0$. Hence $g(x) > 0$ and this completes the proof of the convexity of $f_q(x)$.

From (4b),

$$f_{q-1}(x) = x + P^x f_q(n_q). \tag{B.7}$$

Differentiating twice with respect to x, we obtain

$$f''_{q-1}(x) = f_q(n_q)(\ln P)^2 P^x \tag{B.8}$$

which is non-negative since $f_q(x) > 0$, $x > 0$. By using the same argument, it is easily seen that $f_{q-2}(x)$, $f_{q-3}(x)$, ... , $f_0(x)$ are all convex functions for $x > 0$.

APPENDIX C - CHU'S ANALYSIS OF OPTIMAL BLOCK LENGTH

Chu's analysis of optimal block length for the Stop-And-Wait scheme using end-of-message character [9] is briefly described as follows:

The expected wasted time due to acknowledgement delay and retransmission is :

$$\overline{W}_1(B) = \overline{N}(B)\cdot\overline{A}(B+b) \qquad (C.1)$$

where $\overline{N}(B)$ is the expected number of packets per message and is given as equation (2.2.3) for geometrically distibuted message length
and $\overline{A}(B+b)$ is the expected acknowledgement overhead for a packet size of (B+b). Chu gave $\overline{A}(B+b)$ as:

$$\overline{A}(B+b) = A + \sum_{i=1}^{\infty} (P(B+b))^{i}(A+\frac{B+b}{R}) \qquad (C.2)$$

The expected wasted time due to packet overhead is:

$$\overline{W}_2(B) = N(B)\cdot\frac{b}{R} - \frac{b}{R} \qquad (C.3)$$

Therefore the total expected wasted time per message using the end-of-message character for the Stop-And-Wait scheme is:

$$\overline{W}_c(B) = \overline{W}_1(B) + \overline{W}_2(B) \qquad (C.4)$$

A non explicit equation (10a) of [9] for the optimal block length can thus be obtained by differentiating equation (C.4) with respect to B and set to zero.
It should be noted that equation (C.2) assumes all packets are of the same length (B+b),which is a wrong assumption to be made if an end-of-message character is used in the last packet. Therefore, the above analysis of optimal block length is not exact.

APPENDIX D - EQUALIZATION OF PACKET LENGTHS

A proof of why equalization of packet length can improve the expected wasted time for the Stop-And-Wait scheme using end of message character is given in this appendix. If a message of length ℓ is packetized into blocks of (B+b) bits long, the last packet will consist of $(\ell - \lfloor \ell/B \rfloor B)$ bits long whenever (ℓ/B) is not an integer. For $(\ell - \lfloor \ell/B \rfloor B)$ smaller or comparable to b, the last packet will consist of a large amount of overhead. Therefore, it might be wise to distribute the information bits of the last packet into the $\lfloor \ell/B \rfloor$ blocks so as to equalize the lengths of the packets. These are justified as follows:
Defining f(B) as the time it takes to transmit a packet of length (B+b):

$$f(B) = \left(A + \frac{B+b}{R}\right)\frac{1}{1-P} \qquad (D.1)$$

where P is the block error probability.
For random error channel:

$$f(B) = \left(A + \frac{B+b}{R}\right)\frac{1}{(1-P_b)^{B+b}} \qquad (D.2)$$

By differentiating equation (D.2) twice with respect to B, it is shown that :

$$f''(B) = (1-P_b)^{-(B+b)} \cdot \left(\frac{2}{R} + \left(A + \frac{B+b}{R}\right)\ell n\frac{1}{1-P_b}\right)\ell n\frac{1}{(1-P_b)} \geqslant 0$$

Therefore ,f(B) is a convex function. Jensen's theorem [18] states that if f is a convex function and x_k (k=1,2,...n) never decreases, and if c_k (k=1,2...n) satisfies the conditions:

$$\sum_{k=1}^{n} c_k > 0$$

and for v=1,2..n

$$0 \leqslant \sum_{k=v}^{n} c_k \leqslant \sum_{k=1}^{n} c_k$$

then

$$f\left(\frac{\sum_{k=1}^{n} c_k x_k}{\sum_{k=1}^{n} c_k}\right) \leq \frac{\sum_{k=1}^{n} c_k f(x_k)}{\sum_{k=1}^{n} c_k} \qquad (D.3)$$

If $c_k = 1$ for k=1,2...n

$$f\left(\frac{\sum_{k=1}^{n} x_k}{n}\right) \leq \frac{\sum_{k=1}^{n} f(x_k)}{n} \qquad (D.4)$$

Let

$$x_k = \begin{cases} B & \text{for } k=1,2\ldots n-1 \\ \ell-(n-1)B & \text{for } k=n \end{cases} \qquad (D.5)$$

where $n=\lceil \ell/B \rceil$
Substituting X_K into D.4 for $k=1,2,..n$:

$$f\left(\frac{(n-1)B+\ell-(n-1)B}{n}\right) \leqslant \frac{(n-1)f(B)+f(\ell-(n-1)B\ )}{n}$$

$$n \cdot f\left(\frac{\ell}{n}\right) \leq (n-1)\cdot f(B) + f(\ell-(n-1)B) \qquad (D.6)$$

Equation D.6 shows that the time it takes to transmit (n-1) packets of (B+b) bits and one packets of (ℓ -(n- 1)B+b) bits is always greater or equal to that of transmitting n packets of (ℓ /n+b) bits. This implies that the lengths of the n packets should be equalized as much as possible rather than leaving the last packet as a short packet with large amount of overhead in it. The argument of equalization of block length is thus complete.

APPENDIX E - DERIVATION OF EQUATION (4.2.3)

The derivation of equation (4.2.3) is shown as follows: Differentiating equation (4.2.1) with respect to B and setting the derivative to zero,

$$\frac{\partial W_A}{\partial B} = \frac{\partial}{\partial B}\left(\frac{\ell}{B}\cdot\frac{1}{T}\cdot\frac{B+b}{R} - \frac{\ell+b}{R}\right)=0 \tag{E.1}$$

where $T=(1-P)/S$ (throughput for the Stop-And-Wait scheme)

$S=A\cdot R/(B+b)+1$ (acknowledgement delay in blocks)

$P=1-(1-P_b)^{B+b}$ (assuming random error channel)

Since ℓ is independent of B, equation (E.1) is reduced to :

$$\frac{\partial}{\partial B}\left(\frac{1}{B}\cdot\frac{1}{(1-P_b)^{B+b}}\cdot\left(A+\frac{B+b}{R}\right)\right)=0 \tag{E.2}$$

$$\frac{B^2}{R} + \left(A+\frac{b}{R}\right)B+\left(A+\frac{b}{R}\right)\frac{1}{\ell n(1-P_b)}=0 \tag{E.3}$$

Solving the quadratic equation (E.3) with B>0 ,

$$B^{\infty}= \frac{-\left(A+\frac{b}{R}\right)+\sqrt{\left(A+\frac{b}{R}\right)^2-4\left(A+\frac{b}{R}\right)\frac{1}{R\cdot\ell n(1-P_b)}}}{\frac{2}{R}} \tag{E.4}$$

where

$$\left(A+\frac{b}{R}\right)^2-4\left(A+\frac{b}{R}\right)\frac{1}{R\cdot\ell n(1-P_b)}>0 \tag{E.5}$$

The derivation of equation (4.2.3) is thus completed.

BIBLIOGRAPHY

1. R.D. Stuart, "An insert system for use with feedback communication links," IEEE Trans. Commun. System, vol. CS-11, pp.142-143, March 1963.

2. R.J. Benice and A.H. Frey, Jr.,"An analysis of retransmission systems," IEEE Trans. Commun. System, vol. CS-12,PP.135-144, December 1964.

3. A.R.K. Sastry, "Improving automatic repeat-request (ARQ) performance on satellite channels under high error rate conditions," IEEE Trans. on Commun. ,vol.COM-23,pp.436-439 April 1975.

4. J.M. Morris, "On another go-back-N ARQ technique for high error rate conditions," IEEE Trans. on Commun. , vol. CoM-26,PP.187-189, January 1978.

5. M.J. Miller and S. Lin, "The analysis of some selective-repeat ARQ schemes with finite receiver buffer," IEEE Trans. on Commun. ,vol. COM-29,pp. 1307-1315, September 1981.

6. E.J. Weldon,Jr., "An improve selective-repeat ARQ strategy," IEEE Trans. on Commun., vol. COM-30, pp.480-486, March 1982.

7. A.R.K. Sastry, "Performance of hybrid error control schemes on satellite channels," IEEE Trans. on Commun. vol. COM-23, pp689-694, July 1975.

8. C. Leung and A. Lam,"Forward error correction for an ARQ scheme," IEEE Trans. on Commun. ,vol. COM-29, pp.1514-1519, October 1981.

9. W.W. Chu, "Optimal message block size for computer communications with error detection and retransmission strategies," IEEE Trans. on Commun. ,vol. COM-22, PP. 1516-1525, October 1974.

10. J.M. Morris, "Optimal blocklengths for ARQ error control schemes," IEEE Trans. on Commun. ,vol. COM-27, pp.488-493, Feb. 1979.

11. S.K. Leung and M.E. Hellman, "Concerning a bound on undetected error probabiliy," IEEE Trans. on Info. Theory. , vol. IT-22, NO.2, March 1976.

12. W.W. Peterson and E.J. Weldon, Error-Correcting Codes , M.I.T. Press, 1972.

13. S. Lin, An intoduction to Error-Correcting Codes. ,

Prentice-Hall, 1970.

14. G.A. Arredondo, W.H. Chriss, and E.H. Walker, "A multipath fading simulator for mobile radio," IEEE Trans. on Commun. ,vol. COM-21, Nov. 1973.

15. J.I. Smith, "A Computer generated multipath fading simulation for mobile radio," IEEE Trans. Vehic. Technol., vol. VT-24, August 1975.

16. S.E. Tavares and S,G.S. Shiva, "Detecting and correcting multiple bursts for binary codes," IEEE Trans. Info. Theory. ,vol. IT-16, Sept. 1970.

17. P.J. Mabey, "Mobile radio data transmission-coding for error control," IEEE Trans. Vehic. Technol. vol. VT-27, Aug. 1978.

18. D.S. Mitrinovic, Analytic Inequalities ,Springer-Verlag 1970.

19. K. Mase,T. Takenaka and H. Yamamoto,"Go-Back-N ARQ schemes for point-to-multipoint satellite communications," IEEE Trans. on Commun. , vol. COM-31, April 1983.

20. A. Tanenbaum, Computer Networks , Prentice-Hall, 1981.

21. M. Schwartz, Computer-Communication Network Design and Analysis , Prentice-Hall ,1977.