Titre: Title:	Sequential decoding with ARQ and code combining : a robust hybrid FEC/ARQ system
Auteurs: Authors:	Samir Kallel, & David Haccoun
Date:	1987
Type:	Rapport / Report
Référence: Citation:	Kallel, S., & Haccoun, D. (1987). Sequential decoding with ARQ and code combining : a robust hybrid FEC/ARQ system. (Technical Report n° EPM-RT-87-49). https://publications.polymtl.ca/10076/

Document publié chez l'éditeur officiel
Document issued by the official publisher

Institution:	École Polytechnique de Montréal
Numéro de rapport: Report number:	EPM-RT-87-49
URL officiel: Official URL:	
Mention légale: Legal notice:	

EPM/RT-87/49

SEQUENTIAL DECODING WITH ARQ AND CODE COMBINING: A ROBUST HYBRID FEC/ARQ SYSTEM

SAMIR KALLEL, Ing., M. Sc.A.
DAVID HACCOUN, Ing., Ph.D.

Département de génie électrique
Section communication

Ecole Polytechnique de Montréal

Décembre 1987

Dépôt légal, 4e trimestre 1987
Bibliothèque nationale du Québec
Bibliothèque nationale du Canada

Pour se procurer une copie de ce document, s'adresser au:

Éditions de l'École Polytechnique de Montréal
École Polytechnique de Montréal
Case postale 6079, Succursale A
Montréal (Québec) H3C 3A7
(514) 340-4000

Compter 0,10$ par page (arrondir au dollar le plus près) et ajouter 3,00$ (Canada) pour la couverture, les frais de poste et la manutention. Régler en dollars canadiens par chèque ou mandat-poste au nom de l'École Polytechnique de Montréal. Nous n'honorerons que les commandes accompagnées d'un paiement, sauf s'il y a eu entente préalable dans le cas d'établissements d'enseignement, de sociétés ou d'organismes canadiens.

SEQUENTIAL DECODING WITH ARQ AND CODE COMBINING :
A ROBUST HYBRID FEC/ARQ SYSTEM*

by

Samir Kallel and David Haccoun

Department of Electrical Engineering
École Polytechnique de Montréal
C.P. 6079, Station A
Montréal, Québec, Canada H3C 3A7

* This research was supported in part by the Natural Sciences and Engineering Research Council of Canada and by the Fonds pour la Formation de Chercheurs et l'Aide à la Recherche of Quebec.
Part of this paper will be presented at MILCOM 87, Washington D.C.

SEQUENTIAL DECODING WITH ARQ AND CODE COMBINING : A ROBUST HYBRID FEC/ARQ SYSTEM

by

Samir Kallel and David Haccoun

ABSTRACT

Hybrid FEC/ARQ systems allow reliable and efficient communication. In this paper we consider sequential decoding with ARQ and code combining under the time-out condition. That is, whenever the decoding time of a given packet exceeds some predetermined duration, decoding is stopped and retransmission of the packet is requested. However the unsuccessful packets are not discarded, but are combined with their retransmitted copies. We show that the use of code combining allows sequential decoding to operate efficiently even when the coding rate R exceeds the computational cut-off rate R_{comp}. Furthermore, an analysis of the Selective-Repeat ARQ scheme shows that the use of code combining yields a significant throughput even at very high channel error rates, thus making the system very robust under severe degradations of the channel.

I INTRODUCTION

In order to maintain reliable and efficient communication over noisy channels, hybrid forward-error-correction and automatic-repeat-request (FEC/ARQ) are the techniques communly used. The ARQ system provides the very low undetected error probability performance required, while the FEC system reduces the number of retransmissions by correcting as many packets in error as possible.

In conventional FEC/ARQ schemes, whenever a packet of data needs to be retransmitted, this packet is discarded and replaced by its retransmitted copy. Thus the decoder uses only a single copy of a packet at a time and ignores the information contained in all previous copies. When the channel becomes very noisy, such systems fail to provide a significant throughput. Recently Chase, Mullers and Wolf [1] have suggested a way of combining all the received copies of a packet in order to achieve useful throughput at very high channel error rates. This technique of combining noisy packets is known as code combining [2]. In [1] the FEC system consists of a convolutional encoder with an optimum Viterbi decoder. This idea of using the information contained in all previous copies of a packet has also been investigated using block coding [3]-[4]. In this paper we exploit the idea of code combining with ARQ in conjunction with convolutional coding and sequential decoding.

Sequential decoding is well known for its very good error correcting capabilities. A significant amount of work has been done on hybrid FEC/ARQ schemes using convolutional coding and sequential decoding [5] - [8]. However, when the channel becomes very noisy, these systems fail to provide useful throughput and may even become totally impractical. That is, whenever the coding rate R exceeds the computational cut-off rate R_{comp}, the average computational effort of sequential decoding becomes unbounded. Therefore as the degradation of the channel becomes more severe, successive decoding failures occur and the number of retransmissions tends to become very large. This leads to an overflow of the buffering resources of the system, and hence to a complete breakdown of the communication. We show that code combining allows sequential decoding to operate efficiently even when the coding rate R exceeds R_{comp} yielding a significant throughput under severe degradations of the channel (error rates approaching 50%), thus making the system very robust.

We assume the reader familiar with the basic notions of convolutional coding and sequential decoding. After a brief description of sequential decoding in section II, the principles of code combining with ARQ are given in section III. Sequential decoding with code combining is analyzed in section IV. It is shown that code combining improves both the computational effort and error performance bounds of sequential decoding. Sequential decoding with ARQ under the time-out condition is then analyzed. Tight upper and lower bounds on the average number of transmissions are derived in section V. Applications to the Ideal-Selective. Repeat ARQ scheme is presented in section VI. Numerical evaluations of the theoretical throughput expression are conducted in section VII together with computer simulations. The theoretical results are in agreement with the simulation, confirming the great advantage of using code combining with sequential decoding.

II SEQUENTIAL DECODING

Sequential decoding is a suboptimal decoding procedure for tree codes where only a fraction of the convolutionally encoded tree is explored in the attempt to determine the most likely transmitted path [9]. There are two main sequential decoding algorithms: the Fano algorithm [10] and the Zigangirov-Jelinek (Z-J) or stack algorithm [11]. Assuming a discrete memoryless channel (DMC), the decoder explores the tree one branch at a time and uses the log-likelihood function or Fano symbol metric [10] given by

$$\gamma_j = \log \left[\frac{P(y_j|x_j)}{P(y_j)}\right] - R \tag{1}$$

where x_j is the j^{th} channel input symbol, y_j is the corresponding received symbol and R is the rate of the code. The total metric for a path of length U symbols is then

$$\Gamma_U = \sum_{j=1}^{U} \gamma_j \tag{2}$$

and a sequential decoder will always attempt to search and extend the path having the largest accumulated metric. At the end of the tree the path with the highest metric is accepted as the decoded path. One of the drawbacks of sequential decoding is that it involves a variable number of computations to decode a given packet of data. The distribution of the number of computations to decode one bit is bounded by a Pareto function [12]

$$P\ (C \geqslant N) \leqslant \beta\, N^{-\alpha},\ N \gg 1 \tag{3}$$

where β is a finite constant independent of N and where α, the Pareto exponent, is given by the parametric equation

$$R = \frac{E_0(\alpha)}{\alpha} \tag{4}$$

In (4) $E_0(\alpha)$ is the Gallager function which depends only on the channel.

For a DMC, $E_0(\alpha)$ is given by [13].

$$E_0(\alpha) = -\log \sum_j \Big[\sum_k p(k)\ [p(j|k)]^{\frac{1}{1+\alpha}}\Big]^{1+\alpha} \tag{5}$$

where j and k are the channel output and input letters respectively, and where p(k) is the probability assignment of input letter k. When the Pareto exponent $\alpha \leqslant 1$, the average number of computations to decode one bit is theoretically unbounded. At $\alpha = 1$, $E_0(\alpha)$ is called the computational cut-off rate (Rcomp) of sequential decoding. Hence operation at R = Rcomp represents the practical limit of sequential decoders. Clearly as α gets larger the number of computations to decode one bit decreases, decreasing with it the decoding time of a packet of data.

Occasionally, decoding of a packet may exceed some practical limitations of the decoder and a decoding failure may occur. One can use a feedback channel to eliminate the decoding failures and take full-advantage of the good error correcting capabilities of sequential decoding. Moreover since those packets which are too difficult to decode are often decoded in error, their retransmissions will then lead to an improvment of the error probability. Therefore, sequential decoding in conjunction with a retransmission procedure may be considered as an hybrid FEC/ARQ technique.

Several retransmission procedures are possible. The decoder may limit the time devoted to the decoding of a given packet to some value T_{max}; whenever this time is reached, decoding is stopped, the current packet is discarded and its retransmission is requested [5]-[7]. In another approach the decoder monitors the variations of the metric of the most likely path; if this metric falls below some specified value, decoding is stopped and retransmission of the current packet is requested [7]-[8]. The remaining available memory of the decoder may also be used as a criteria to request retransmissions [8].

In this paper we consider only the first retransmission procedure, i.e. the time-out condition. Assuming the decoding time proportional to the number of computations, then if the total number of computations C_L performed for the decoding of a given packet exceeds some value C_{max}, decoding is stopped and retransmission of that packet is requested. This decoding failure event occurs with probability P(F) given by

$$P(F) = P(C_L \geqslant C_{max}) \tag{6}$$

For an L-bit information packet, the probability of a decoding failure P(F) can be approximated from (3) [7], [14] as

$$P(F) \simeq L \; \beta \; C_{max}^{-\alpha} \tag{7}$$

III CODE COMBINING IN ARQ SYSTEMS

Code combining is a technique for combining any number of repeated packets encoded with a rate-R code, in order to obtain a lower coding rate, and thus a more powerful error-correcting code [1].

In ARQ systems using code combining, whenever a packet needs to be retransmitted, this packet is not discarded as in usual ARQ. Upon receiving the retransmitted copy, the decoder interlaces the two copies and reprocesses the two combined copies as a packet issued from a 2-repetition-code of rate R/2 bits/symbol. Should a second retransmission be requested, then as the retransmitted copy becomes available, the decoder combines it with the two earlier ones, and again reprocesses the combined packet as a packet issued from a 3-repetition-code of rate R/3 bits/symbol. This procedure is continued until decoding succeeds.

IV SEQUENTIAL DECODING WITH FIXED NUMBER OF REPEATS AND CODE COMBINING

In this section, we assume a given encoded packet is repeated n times. The n received copies are combined and presented to a sequential decoder. The operations of sequential decoding with n-repetition-code combining are discussed and its performances analysed.

Let $\underline{I} = (I_1, I_2, \ldots, I_L)$ be an L-bit information packet encoded into the sequence $\underline{X} = (x_1, x_2, \ldots x_U)$. Let the encoded packet $\underline{X}$ be repeated n times and let the n repetitions be transmitted over a DMC* consecutively. Let the n received sequences be

*The DMC may be different for each repetition. A transition probability set is then considered at each repetition.

$$\underline{Y}^{(i)} = (y_1^{(i)}, y_2^{(i)}, \dots y_U^{(i)}), \ i = 1,2, \dots, n \tag{8}$$

The decoder interlaces the sequences $\underline{Y}^{(i)}$, i=1, 2, ..., n, and forms the sequence

$$\underline{Y} = (y_1^{(1)}, y_1^{(2)}, \dots, y_1^{(n)}, \dots, y_j^{(1)}, y_j^{(2)}, \dots, y_j^{(n)}, \dots, y_U^{(1)}, y_U^{(2)} \dots, y_U^{(n)}) \tag{9}$$

Each received group of n letters $(y_j^{(1)}, y_j^{(2)}, \dots, y_j^{(n)})$ corresponds to the n transmissions of the same encoded symbol x_j. Those n letters are thus statistically dependent and hence the sequential decoder must consider them as a group of n letters at a time rather than as a series of single letters.

We can model the encoder and the n-repeater as a single device. Instead of repeating n times the packet $\underline{X}$, let us repeat n times each symbol of $\underline{X}$. The resulting sequence denoted by $\underline{X}^{(n)}$ is thus given by

$$\underline{X}^{(n)} = (\underbrace{x_1, x_1, \dots, x_1}_{\text{n repetitions}}, \dots, \underbrace{x_j, x_j, \dots, x_j}, \dots \underbrace{x_U, x_U, \dots, x_U}) \tag{10}$$

If this sequence $\underline{X}^{(n)}$ were transmitted over a DMC, it would produce the same sequence $\underline{Y}$ given by (9), where for each n-repetition symbol $(x_j, x_j, \dots, x_j)$ corresponds the n letters $(y_j^{(1)}, y_j^{(2)}, \dots, y_j^{(n)})$.

Considering each n-repetition symbol $(x_j . x_j, \dots, x_j)$ as an entity, the convolutional encoder, of rate R = b/V bits/symbol, followed by the n-repeater, may be viewed as an "apparent convolutional encoder". For each b-information bit input, the apparent encoder produces V output groups of n-repetition symbol. The rate of this apparent encoder is thus R_n = b/V bits/n-repetition symbol, and hence we have

$$R_n = R \tag{11}$$

The channel used for the n consecutive repetitions may also be viewed as an "apparent channel" producing n output letters $(y_j^{(1)}, y_j^{(2)}, \ldots, y_j^{(n)})$ for each input n-repetition symbol $(x_j, x_j, \ldots, x_j)$. This apparent channel is clearly a DMC.

With the above modelling as a single apparent encoder and as a single apparent channel, we can now apply known sequential decoding results to sequential decoding with n-repetition-code combining.

A. Fano Metric for Sequential Decoding with n-Repetition-Code Combining

Substituting in (1) each symbol x_j by an n-repetition symbol $(x_j, x_j, \ldots, x_j)$, each symbol y_j by an n-tuple letters$(y_j^{(1)}, y_j^{(2)}, \ldots, y_j^{(n)})$, and using (11), the n-repetition symbol or Fano metric is now given by

$$\gamma_j^{(n)} = \log \left[\frac{\prod_{i=1}^{n} P(y_j^{(i)} \mid x_j)}{P(y_j^{(1)}, y_j^{(2)}, \ldots, y_j^{(n)})} \right] - R \tag{12}$$

For example, for the BSC shown on Figure 1.a, with a transition probability p, the corresponding apparent channel for n=2 repetitions is shown on Figure 1.b. For the BSC, the two possible symbol metrics used in sequential decoding with coding rate R are given by

$$\begin{aligned} \gamma_1^{(1)} &= \log 2p - R \quad ; \\ \gamma_2^{(1)} &= \log 2\,(1-p) - R \end{aligned} \tag{13}$$

When sequential decoding is used with 2-repetition-code combining, from the apparent channel shown on Figure 1.b, there are now three possible 2-repetition symbol metrics, given by

$$\gamma_1^{(2)} = \log 2 \frac{p^2}{(1-p)^2+p^2} - R \quad ;$$

$$\gamma_2^{(2)} = - R \quad ; \qquad (14)$$

$$\gamma_3^{(2)} = \log 2 \frac{(1-p)^2}{(1-p)^2 + p^2} - R$$

B. Computational Distribution of Sequential Decoding with n-Repetition-Code Combining

In accordance with our model of both the apparent encoder and the apparent channel, the number of computations per decoded bit $C^{(n)}$ of sequential decoding with n-repetition-code combining still follows the usual Pareto distribution but with a substantially reduced variability. Here a computation is still defined as in [12], that is the extension of a node one step further into the tree of the code. We can therefore write

$$P(C^{(n)} \geq N) \leq \beta_n N^{-\alpha_n}, \quad N \gg 1 \qquad (15)$$

where β_n is a finite constant independant of N, and where α_n is the apparent Pareto exponent relative to the n-repetition apparent channel, given by the parametric equation

$$R = \frac{E_o^{(n)}(\alpha_n)}{\alpha_n} \qquad (16)$$

where R is the rate of the original code in bits/symbol and where $E_o^{(n)}(\alpha)$ is the Gallager function of the apparent channel given by

$$E_o^{(n)}(\alpha) = -\log \sum_{j_1} \sum_{j_2} \dots \sum_{j_n} \left[\sum_k P(k) \left[\prod_{i=1}^{n} P(j_i|k) \right]^{\frac{1}{1+\alpha}} \right]^{1+\alpha} \tag{17}$$

The letters k and j refer to the channel input and output symbols respectively and where n is the number of repetitions of each symbol. Clearly for n = 1, (17) corresponds to the usual Gallager Function (5).

Even though a Pareto distribution still prevails, the advantage of using code combining on the variability of the computational effort of sequential decoding appears as an increase of the apparent Pareto exponent, and hence as a reduction of the computational variability. Using the following Lemma, this is proven in the theorem below and its corollary.

Lemma

The Gallager function relative to a n-repetition apparent channel monotonely increases with n.

$$E_o^{(n)}(\alpha) \geq E_o^{(j)}(\alpha) \quad , \quad n \geq j$$

The proof of this lemma is given in the appendix.

Theorem

The apparent Pareto exponent α_n of the distribution of the computational effort of sequential decoding with n-repetition-code combining monotonely increases with n.

$$\alpha_n \geq \alpha_j \ , \ n \geq j \tag{18}$$

Proof

It is known that $E_0(\alpha)$ is a non-decreasing function of α [13]. Now from the lemma above $E_0^{(n)}(\alpha)$ increases monotonely with n, therefore from (16) α_n also increases monotonely with n.

The Pareto distribution of sequential decoding decreases with α. In (15) the constant β_n is independent of N and is less than 1. Therefore as a corollary to the theorem above we can write

Corollary 1

The variability of the computational effort of sequential decoding with n-repetition-code combining decreases with n.

C. Error Probability of Sequential Decoding with n-Repetition-Code Combining

Following [12], the error probability $P^{(n)}(\varepsilon)$ of sequential decoding with n-repetition-code combining is bounded by

$$P^{(n)}(\varepsilon) \leq \begin{cases} A_n \exp\left[-bK\, \alpha_n\, R\right], & R \leq E_0^{(n)}(1) \\ A_n \exp\left[-bK\, E_0^{(n)}(1)\right], & R \geq E_0^{(n)}(1) \end{cases} \qquad (19)$$

Where A_n is a finite constant independent of K, bK is the constraint length of the code, R is the rate of the original code, α_n is the apparent Pareto exponent given by (16), and $E_0^{(n)}(1)$ is the value at $\alpha = 1$ of the Gallager function of the n-repetition apparent channel given by (17).

The error probability of sequential decoding decreases with α or with $E_0(1)$ depending on whether $R \leqslant E_0(1)$, or $R \geqslant E_0(1)$. In (19) the constant A_n is independent of K and is smaller than 1. Therefore from (19) and as a corollary to the above lemma and theorem we can write

Corollary 2

The error probability of sequential decoding with n-repetition code combining decreases with n.

Figure 2 shows the apparent Pareto exponent α_n as a function of E_S/N_0, the energy per channel input symbol-to noise ratio for n = 1 to 9 repetitions over a binary symmetric channel (BSC) and for a code rate R = ½. Figure 1 shows that as n increases the apparent Pareto exponent increases considerably. The range of E_S/N_0 values over which $\alpha_n \geqslant 1$ is therefore increased, making sequential decoding possible for those values of E_S/N_0 over which it would be otherwise precluded. For example when E_S/N_0 = -0.85 (dB) and n = 1, α_1 is equal to 0.1; sequential decoding is clearly not practical for this value of α. However, when n = 2 the apparent Pareto exponent α_2 becomes equal to 1.3 making sequential decoding feasible. Figure 3 shows the distribution of the number of computations per decoded bit obtained from simulation using the stack algorithm for E_S/N_0 = -0.85 (dB) and n = 2.

We have shown that when more than a single copy of a packet are available at the receiver, the use of code combining may be very helpful in sequential decoding. It increases the apparent Pareto exponent, improving thereby the sequential decoder behavior. This situation of having more than a single copy of a packet at the receiver is encountered in ARQ systems where a clever decoder does not discard those noisy packets that must be repeated but combines them with their retransmitted copies.

In the next section we analyze ARQ systems with code combining using sequential decoding under the time-out condition.

V SEQUENTIAL DECODING WITH ARQ AND CODE COMBINING

We consider now sequential decoding with retransmissions under the time-out condition. Assuming the decoding effort proportional to the number of computations, then whenever the total number of computations exceeds some value C_{max}, decoding is stopped and retransmission of the packet is requested. However, the packet is not discarded, but combined with its retransmitted copy. Should decoding still fail, a retransmission is requested again and the three copies of the same packet are combined together and decoding is attempted anew. This procedure of retransmission-code combining is continued until decoding succeeds. The combined packets are decoded in the same manner as explained in the previous section .

Let $F^{(n)}$ and $S^{(n)}$ denote the events "decoding failure" and "decoding success" respectively, when decoding n combined copies of a given packet. We can write

$$P(F^{(n)}) = 1-P(S^{(n)}) = P(C_L^{(n)} \geqslant C_{max}) \tag{20}$$

where $C_L^{(n)}$ is the total number of computations of sequential decoding with n-copies-code combining.

From (7) and (15) the probability of a decoding failure $P(F^{(n)})$ can be approximated by

$$P(F^{(n)}) \simeq L\, \beta_n\, C_{max}^{\;-\alpha_n} \tag{21}$$

The average number of transmissions of a given packet Tr is given by

$$Tr = 1\ P(S^{(1)}) + 2\ P\ (F^{(1)},S^{(2)}) + 3\ P\ (F^{(1)},F^{(2)},S^{(3)}) + \\ \ldots + nP\ (F^{(1)},F^{(2)},\ldots,F^{(n-1)},S^{(n)}) + \ldots \qquad (22)$$

We can express (22) as

$$Tr = 1\ P(S^{(1)}) + 2P(F^{(1)},S^{(2)}) + 2P(F^{(1)},F^{(2)}) - 2P(F^{(1)},F^{(2)}) \\ + 3P(F^{(1)},F^{(2)},S^{(3)}) + 3P(F^{(1)},F^{(2)},F^{(3)}) - 3P(F^{(1)},F^{(2)},F^{(3)}) \\ + \ldots + nP(F^{(1)},F^{(2)},\ldots F^{(n-1)},S^{(n)}) + nP(F^{(1)},F^{(2)},\ldots F^{(n-1)},F^{(n)}) \\ - nP(F^{(1)},F^{(2)},\ldots, F^{(n-1)},F^{(n)}) + \ldots \qquad (23)$$

Rearranging the terms in (23) we obtain

$$Tr = 1 + P(F^{(1)}) + P(F^{(1)},F^{(2)}) + P(F^{(1)},F^{(2)},F^{(3)}) \\ + \ldots + P(F^{(1)},F^{(2)},\ldots,F^{(n)}) + \ldots \qquad (24)$$

Given a decoding failure at the $(j-1)^{th}$ attempt, the probability of a decoding failure at the j^{th} attempt is clearly larger than the unconditional probability of having a failure when decoding j independent combined copies of a packet, that is

$$P(F^{(j)} \mid F^{(j-1)}) \geqslant P(F^{(j)}) \qquad (25)$$

Now, using the fact that

$$P(F^{(1)},F^{(2)},\ldots,F^{(j)}) \leqslant P(F^{(j)}) \qquad (26)$$

we can therefore upper and lower bound Tr as

$$1 + \sum_{n=1}^{\infty} \prod_{i=1}^{n} P(F^{(i)}) < Tr < 1 + \sum_{n=1}^{\infty} P(F^{(n)}) \tag{27}$$

In order to test the tightness of these two bounds, we have evaluated them using the expression of $P(F^{(n)})$, given by (21), where the value of the constant β_n was fixed to 1. Results are given for a packet of size 500 information bits (a typical choice for sequential decoding), a coding rate $R = \frac{1}{2}$, and for a BSC.

Figure 4 shows the variation of the lower and upper bounds on Tr as a function of Cmax/L, for five values of signal to noise ratio Es/No = -6.0, -4.0, -2.0, 0.0 and 2.0 (dB). Figure 5 shows the variation of the two bounds as a function of Es/No, for a fixed value of Cmax/L = 2. On both figures, we can observe that the lower and upper bounds are very nearly identical, indicating that the bounds (27) are indeed very tight.

This behaviour of the lower and upper bounds may be explained as follows. For large values of Es/No, or equivalently for $\alpha > 1$, the dominant term in both summations of (27) is $1 + P(F^{(1)})$, the other terms are negligible; thus both summations yield the same values. Each term $(P(F^{(1)}, F^{(2)},\ldots,F^{(j)})$ in (24) can be written as

$$P(F^{(1)},F^{(2)},\ldots,F^{(j)}) = P(F^{(1)}|F^{(2)},\ldots,F^{(j)})P(F^{(2)}|F^{(3)},\ldots,F^{(j)}) \ldots$$

$$\ldots P(F^{(j-1)}|F^{(j)})\ P(F^{(j)}) \tag{28}$$

The term $P(F^{(j-1)}|F^{(j)})$ is the probability of having a decoding failure when (j-1) copies of a packet are combined together given a decoding failure has occurred when those (j-1) copies are further combined with an additional copy. Now for small values of Es/No, that is for $\alpha_j \leq 1$, the decoding of j-combined copies of a packet fails with a high probability. Conversely, if $\alpha_j > 1$, the decoding of j-combined copies of a packet fails with a small probability. Now from the theorem above, we have $\alpha_{j-1} < \alpha_j$. Therefore, given a decoding failure when j copies of a packet are combined, then if only (j-1) of those j copies are combined together, a decoding failure will occur with high probability, that is

$$P(F^{(j-1)}|F^{(j)}) \simeq 1 \tag{29}$$

Moreover, since $\alpha_{j-1} < \alpha_j < 1$, we can assert that with a high probability, we have

$$P(F^{(j-1)}|F^{(j)}) \simeq P(F^{(j-1)}) \simeq 1 \tag{30}$$

The above discussion can be generalized to all conditional probability terms in (28). Therefore, we can write (28) as

$$P(F^{(1)},F^{(2)},\ldots,F^{(j)}) \simeq \prod_{i=1}^{j} P(F^{(i)}) \simeq P(F^{(j)}) \tag{31}$$

and hence, using (31), both summations of (27) yield nearly the same values.

As a consequence of the tightness of the lower and upper bounds, Tr can be very well approximated by either bound.

Without code combining, successive decoding attempts of a given packet are statiscally independent. The average number of transmissions Tr* is then given by the summation on the left side of (27), where $P(F^{(i)})$ is substituted by $P(F^{(1)})$. The summation then yields

$$Tr^* = \frac{1}{1 - P(F^{(1)})} \tag{32}$$

VI THROUGHPUT FOR THE IDEAL-SELECTIVE-REPEAT ARQ SCHEME IN CONJUNCTION WITH CODE COMBINING

In this section we consider sequential decoding under the time-out condition with the Ideal-Selective-Repeat ARQ scheme in conjunction with code combining. A noiseless feedback channel is assumed. In an Ideal-Selective-Repeat ARQ scheme, a packet is retransmitted only if requested [15].

In sequential decoding, the time required to decode one information bit is not constant. The average throughput may be defined [7] as

$$\Theta = \frac{1}{t_A} \text{ bits/sec} \tag{33}$$

where t_A is the average time required to successfully receive an information bit. However, this definition of the throughput is meaningful as long as the decoder speed is such that it never stays idle. Hence it is the decoder that imposes the rate at which information bits are delivered to the user. But if this were not the case, that is if the decoder speed is so high that it is waiting most of the time for a new packet to process, then the rate at which information bits are delivered to the user is imposed by the channel. In the latter case, a more meaningful definition of the throughput would be the throughput efficiency given by

$$\eta = \frac{R}{Tr} \; \frac{L}{L + (K-1)} \text{ bits/symbol} \tag{34}$$

where the factor $\frac{L}{L + (K-1)}$ is the loss in throughput due to the tail of (K-1) known bits appended to each packet. This factor may be ignored if $(K-1) \ll L$.

A more precise definition of the throughput takes into consideration both the channel transmission rate and the decoder speed [6].

In this paper we assume that the decoder speed is not a limiting factor on the throughput. The second definition of the throughput, given by (34) is thus adopted. This assumption is not unrealistic since in practice it is desirable to have a decoder speed high enough so that eventual overflows of the decoder input buffer are minimized.

The computation of the throughput can be conducted by the use of the expression of $P(F^{(n)})$ given by (21). Now since in (21) the constant $\beta_n \leqslant 1$, it can be approximated by 1 and hence a lower bound on the throughput can be calculated.

VII NUMERICAL AND SIMUALTION RESULTS

Computer simulation with the stack algorithm has been used in order to verify the theoretical results. The code used was a rate ½ optimum distance profile systematic code of constraint length 24, defined by its octal generator 67 11 51 43 [16]. The data is organized in packets of length 500 information bits to which a tail of 23 known bits is appended. For every run, the simulation was continued until at least 1000 packets had been decoded and accepted, and at least 25 retransmission requests had been made.

Selection of C_{max}/L value

In both computer simulation and numerical evaluation of the throughput the ratio C_{max}/L must be properly chosen. As indicated by (21), the decoding failure probability decreases as C_{max}/L increases. Thus, one may be tempted to choose a large value of C_{max}/L in order to maximize the throughput. However, letting C_{max}/L becoming too large involves a large and wasted decoding time before a retransmission is requested. This may allow a queue to form at the input buffer of the decoder, leading eventually to its overflow. Moreover as we mentioned earlier, packets requiring too much decoding time are often decoded in error. Hence a large value of C_{max}/L may also degrade the error performance. A reasonable choice of C_{max}/L would be slightly larger than the average number of computations per decoded bit in normal operation of sequential decoding ($\alpha \geqslant 1$). With a such choice, in good channel conditions, decoding of a packet proceeds normally. On the other hand, when the channel becomes very noisy, a reasonable value of C_{max}/L will prevent the decoder from wasting too much time attempting to decode packets that will be most probably retransmitted. This procedure will prevent possible overflows of the input buffer of the decoder and possibly also decoding errors. In the sequel all results are given for a value of $C_{max}/L = 2.0$.

Numerical values and simultion results for the throughput are presented in figure 6. We can observe that the theoretical values of the throughput closely agree with the simulation results. For large values of E_S/N_o, $E_S/N_o \geqslant 1.6$ (dB), corresponding to $R \leqslant R_{comp}$ ($\alpha \geqslant 1$), code combining does not yield a throughput improvement over ARQ alone. However as the channel degrades, the throughput without code combining drops rapidly to zero, whereas with code combining a significant throughput is still achieved even at very small values of E_S/N_o.

Figure 7 compares the throughput of fixed rates 1/4, 1/6, 1/8 decoding without code combining with a rate 1/2 decoding with code combining. In certain ranges of E_S/N_0, low rates decoding yields a slightly better throughput than rate 1/2 decoding with code combining. However the difference is relatively small, with the advantage of rate 1/2 decoding with code combining of being adaptive to channel conditions. This is a very interesting point of code combining. In some situations, selecting the appropriate coding rate to use may be a difficult problem. With code combining an arbitrary coding rate may be chosen to start with, yielding a very little throughput degradation.

Figure 8 shows the variations of the throughput as a function of E_S/N_0 for different coding rates with the use of code combining. We can observe that code combining makes the use of high coding rates attractive in varying channel conditions. For large values of E_S/N_0 a system with a high coding rate yields a better throughput performance over a system with a low coding rate. However, for smaller values of E_S/N_0, code combining makes the throughput degradation relatively small.

VII CONCLUSION

In this paper we analyzed sequential decoding with ARQ and code combining under the time-out condition. We have shown that the use of code combining increases the Pareto exponent at each subsequent decoding attempt of the same packet, thus allowing sequential decoding to operate efficiently even when the coding rate R exceeds the computational cut-off rate R_{comp}. Using an approximate expression for the decoding failure probability, we have derived tight upper and lower bounds on the average number of transmissions. Application to the Ideal-Selective-Repeat ARQ scheme has shown that at $R \leqslant R_{comp}$ the use of code combining does not yield a throughput improvement over ARQ alone. However, as the channel degrades, the throughput without code combining drops rapidly to zero, whereas, with code combining a significant throughput is still achieved even at very high channel error rates. Therefore, code combining allows the system to be adaptive to channel conditions, making the use of high coding rates attractive even under severe channel degradations. Furthermore, since the use of code combining reduces the average number of retransmissions, the probability of an overflow of the buffering resources of the system is also reduced, thus making the system very robust. Computer simulation have consistently confirmed these theoretical results.

APPENDIX

We prove in this appendix that

$$E_o^{(n)}(\alpha) \geq E_o^{(j)}(\alpha) \quad , \quad n \geq j \tag{A1}$$

Proof

To prove (A1), one needs to prove that

$$E_o^{(n)}(\alpha) \geq E_o^{(n-1)}(\alpha) \tag{A2}$$

from (14)

$$E_o^{(n)}(\alpha) = -\log \sum_{j_1} \sum_{j_2} \cdots \sum_{j_n} \left[\sum_k p(K) \left[\prod_{i=1}^{n} p(j_i|k)\right]^{\frac{1}{1+\alpha}}\right]^{1+\alpha}$$

Using the inequality (variant of Minkowski inequality) [17] p. 199

$$\sum_j \left[\sum_k Q_k \left[a_{kj}\right]^\lambda\right]^{\frac{1}{\lambda}} \leq \left[\sum_k Q_k \left[\sum_j a_{kj}\right]^\lambda\right]^{\frac{1}{\lambda}} \tag{A3}$$

in which we subtitute,

j by j_n, Q_k by $p(k)$, a_{kj} by $\prod_{i=1}^{n} P(j_i|k)$ and λ by $1 / 1+\alpha$

we can then write

$$\sum_{j_n} \left[\sum_{k} p(k) \left[\prod_{i=1}^{n} p(j_i|k)\right]^{\frac{1}{1+\alpha}}\right]^{1+\alpha} \leq$$

$$\left[\sum_{k} p(k) \left[\sum_{j_n} \left[\prod_{i=1}^{n} p(j_i|k)\right]^{\frac{1}{1+\alpha}}\right]\right]^{1+\alpha}$$

and therefore we have

$$E_o^{(n)}(\alpha) \geq -\log \sum_{j_1} \sum_{j_2} \cdots \sum_{j_{n-1}} \left[\sum_{k} p(k) \left[\sum_{j_n} \prod_{i=1}^{n} p(j_i|k)\right]^{\frac{1}{1+\alpha}}\right]^{1+\alpha}$$

$$= -\log \sum_{j_1} \sum_{j_2} \cdots \sum_{j_{n-1}} \left[\sum_{k} p(k) \left[\prod_{i=1}^{n-1} p(j_i|k)\right]^{\frac{1}{1+\alpha}}\right]^{1+\alpha}$$

$$= E_o^{(n-1)}(\alpha)$$

Q.E.D.

REFERENCES

[1] D. Chase, P.D. Mullers and J.K. Wolf, "Application of code combining to a Selective-Repeat ARQ Link", Proceedings of MILCOM 1985.

[2] D. Chase "Code Combining. A Maximum-Likelihood Decoding Approach for Combining an Arbitrary Number of Noisy Packets", IEEE Transactions on Communications, vol. Com-33, May 1985.

[3] C. Lau and C. Leung, "Performance Analysis of a Memory ARQ Scheme with Soft Decision Detectors", IEEE Transactions on Communications, Vol. Com-34, August 1986.

[4] P.S. Sindhu, "Retransmission Error Control with Memory", IEEE Transactions on Communications, Vol. Com-25, May 1977.

[5] N. Shacham, "Performance of ARQ with sequential Decoding Over One-Hop and Two-Hop Radio Links", IEEE Transactions on Communications, Vol. Com-31, October 1983.

[6] N. Shacham, "ARQ with Sequential Decoding of Packetized Data: Queuing Analysis", IEEE Transactions on Communications, Vol. Com-32, October 1984.

[7] A. Drukarev and D.J. Costello, Jr., "Hybrid ARQ Error Control Using Sequential Decoding", IEEE Transactions on Information Theory, Vol. IT-29, July 1983.

[8] P.Y. Pau and D. Haccoun, "An Analysis of Sequential Decoding with Retransmission Procedures", Technical Report, EMP/RT-85-19, École Polytechnique de Montréal, May 1985.

[9] V.K. Bhargava, D. Haccoun, R. Matyas and P. Nuspl, "Digital Communication by Satellite", John Wiley and Sons, Inc., 1981.

[10] R.M. Fano, "A Heuristic Discussion of Probabilistic Decoding", IEEE Transactions on Information Theory, Vol. IT-9, April 1963.

[11] F. Jelinek, "Fast Sequential Decoding Using a Stack", IBM Journal of Research and Development, Vol. 13, November 1969.

[12] G.D. Forney, "Convolutional Codes, III. Sequential Decoding", information and control, Vol. 25, 267-297, July 1974.

[13] R.G. Gallager, "Information Theory and Reliable Comminication", John Wiley and Sons, Inc., 1968.

[14] I.M. Jacobs and E.R. Berlekamp, "A Lower Bound to the Distribution of Computation for Sequential Decoding" IEEE Transactions on Information Theory, Vol. IT-13, April 1967.

[15] S. Lin and D.J. Costello, Jr., "Error Control Coding: Fundamentals and Applications", Prentice-Hall, Inc., 1983.

[16] Johannesson, R., "Robustly Optimal Rate One-Half Binary Convolutional Codes", IEEE Transactions on Information Theory, Vol. IT-21, July 1975.

[17] Viterbi, A.J. and Omura, J.K., "Principles of Digital Communication and Coding", Mc Graw-Hill, 1979.

LIST OF FIGURES

Figures

1a : Binary symmetric channel.

1b : 2-Repetition apparent channel.

2 : n-Repetition apparent Pareto exponent.

3 : Empirical distribution of the number of computations per decoded bit of sequential decoding with 2- repetition-code combining.

4 : Variation of the lower and upper bounds on the average number of transmissions Tr as a function of C_{max}/L.

5 : Variation of the lower and upper bounds on the average number of transmissions Tr as a function of E_S/N_o .

6 : Numerical and simulation results for the throughput.

7 : Throughput comparison of fixed rates 1/4, 1/6, 1/8 decoding without code combining, with a rate 1/2 decoding using code combining.

8 : Throughput for different coding rates with code combining.

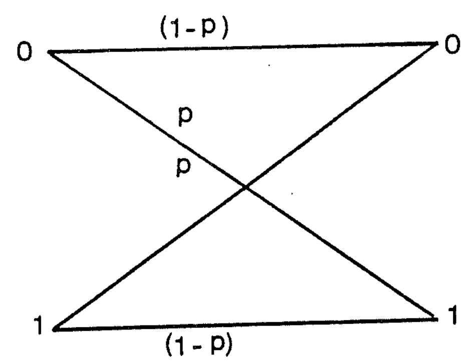

Figure 1a : Binary symmetric channel

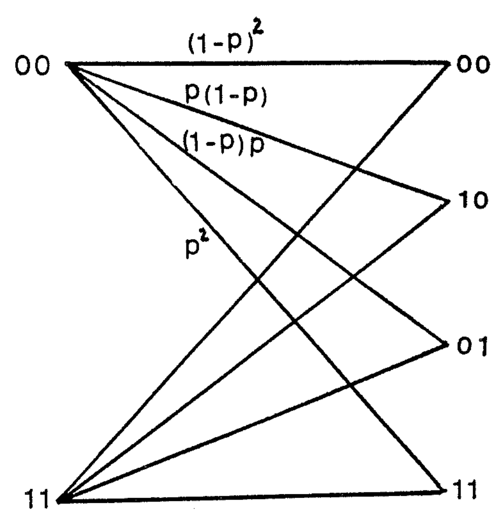

Figure 1b : 2-Repetition apparent channel

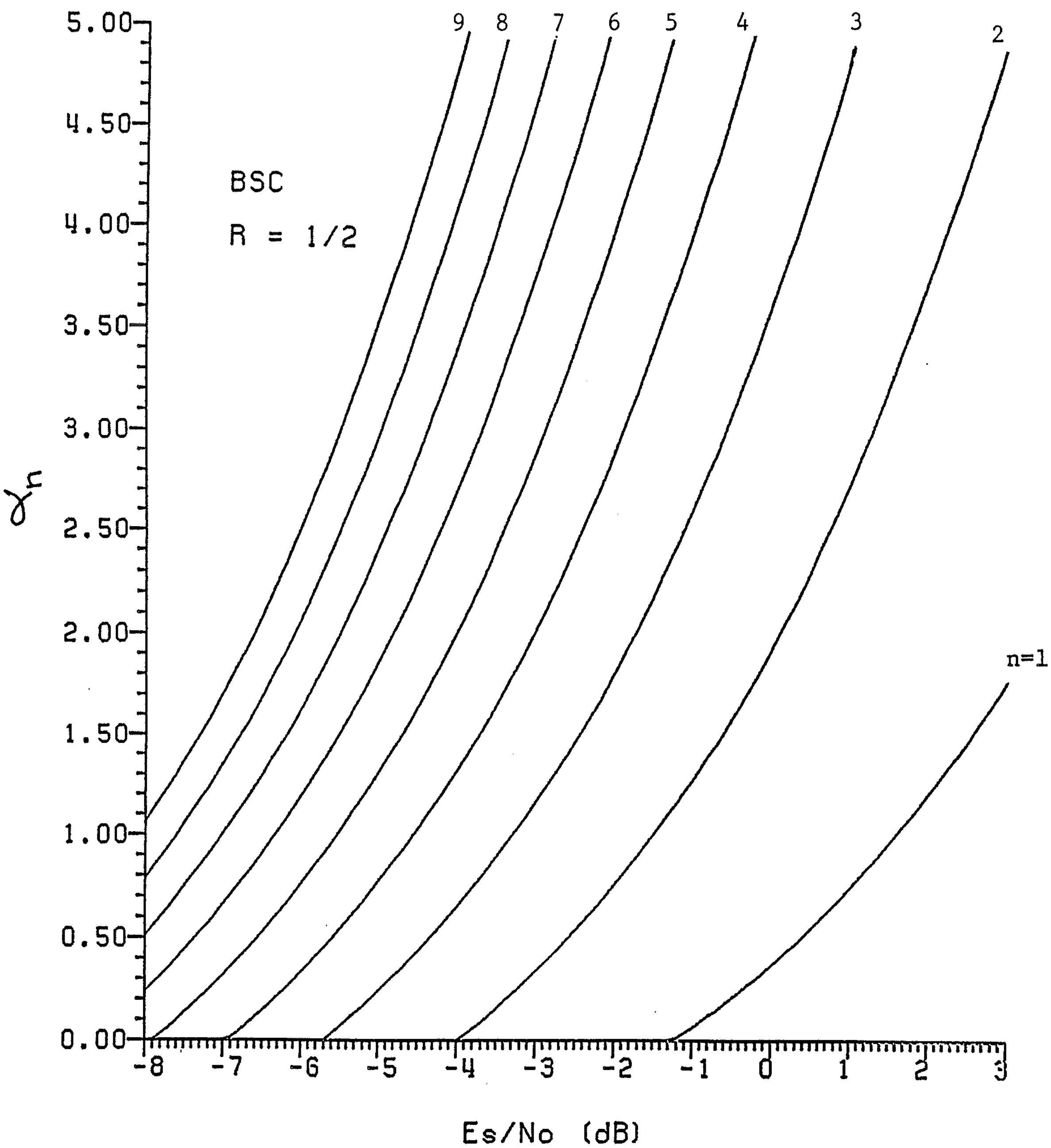

Figure 2 : n-Repetition apparent Pareto exponent

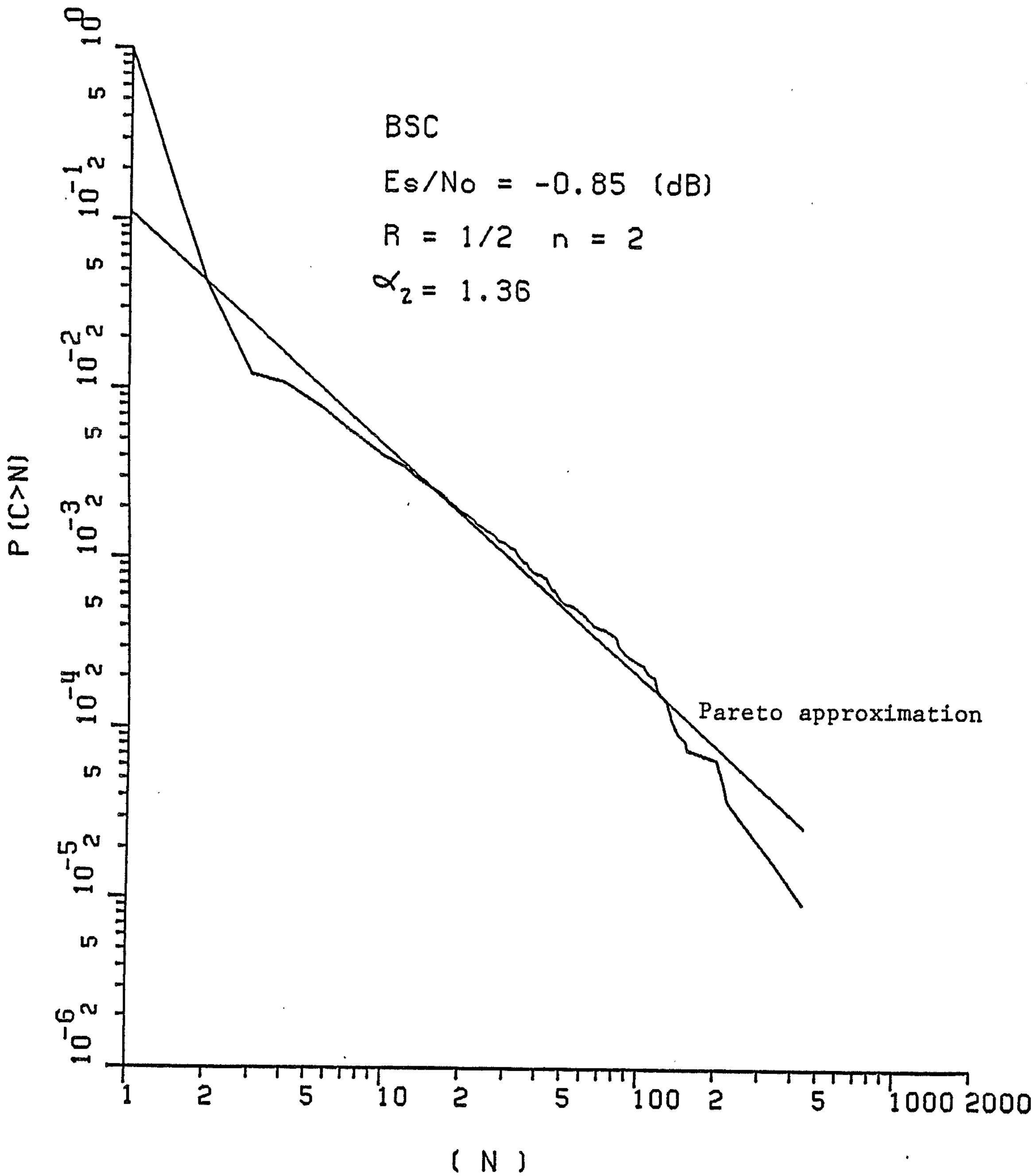

Figure 3 : Empirical distribution of the number of computations per decoded bit of sequential decoding with 2-repetition-code combining

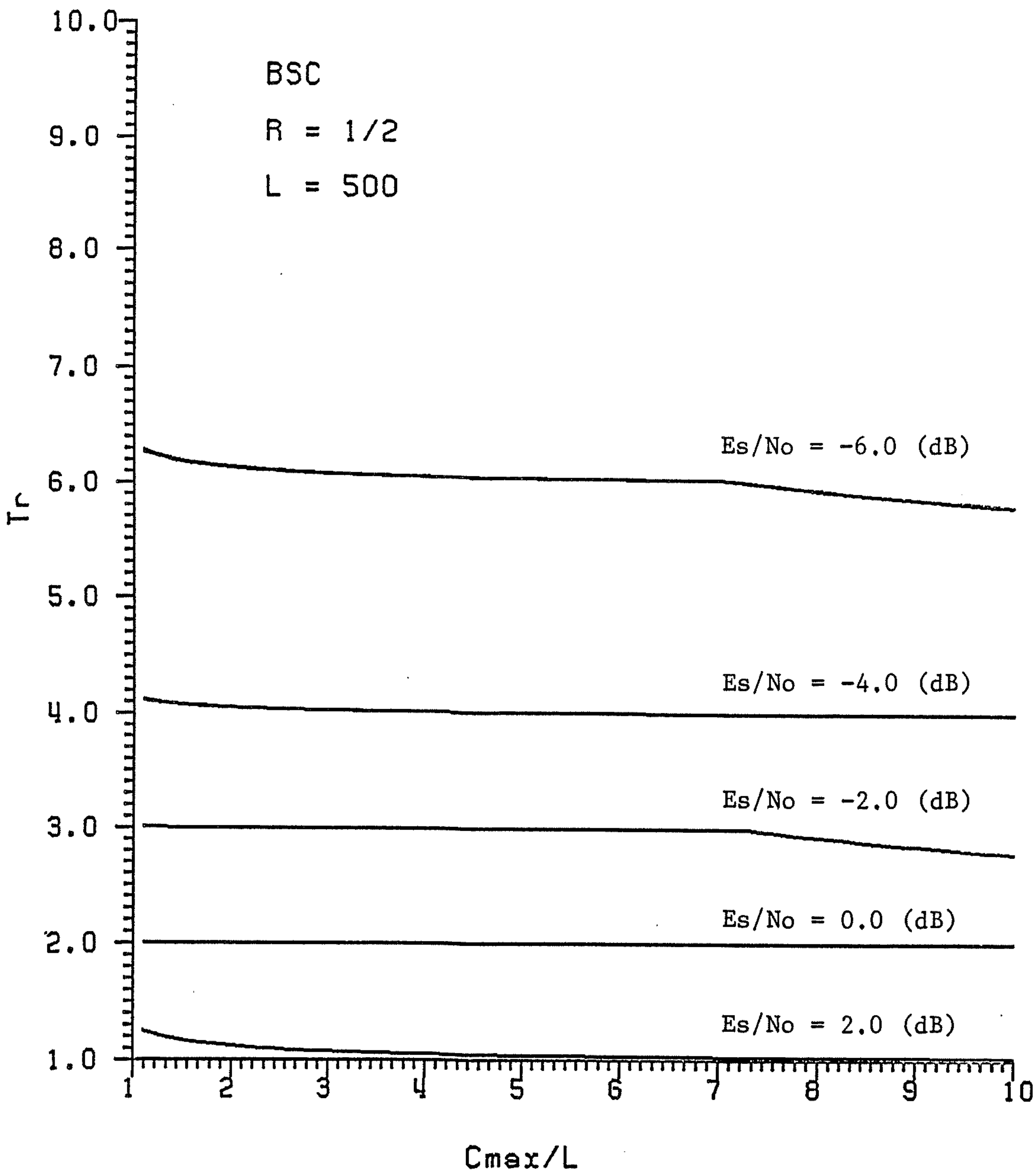

Figure 4 : Variation of the lower and upper bounds on the average number of transmissions Tr as a function of C_{max}/L

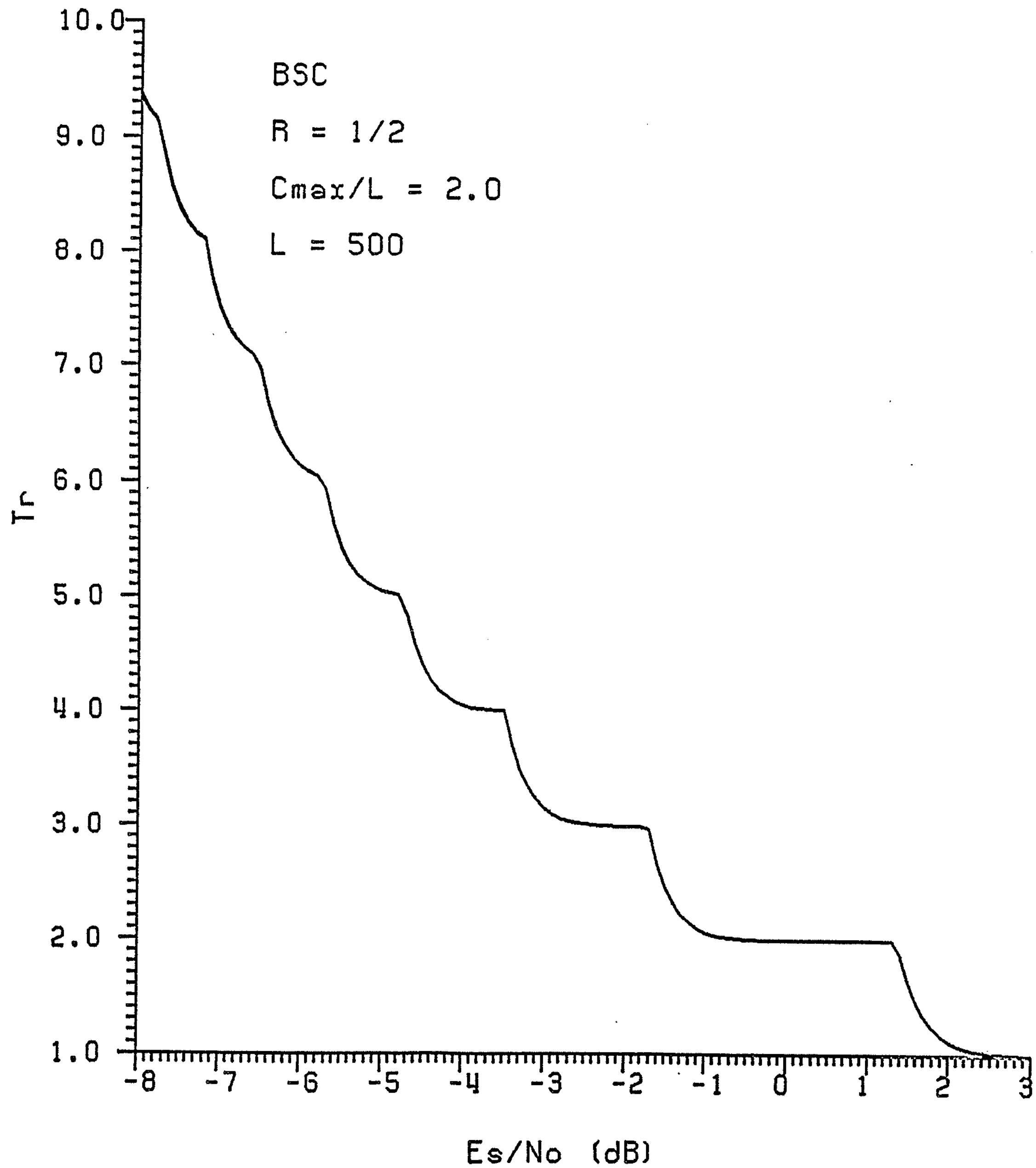

Figure 5 : Variation of the lower and upper bounds on the average number of transmissions Tr as a function of ES/No

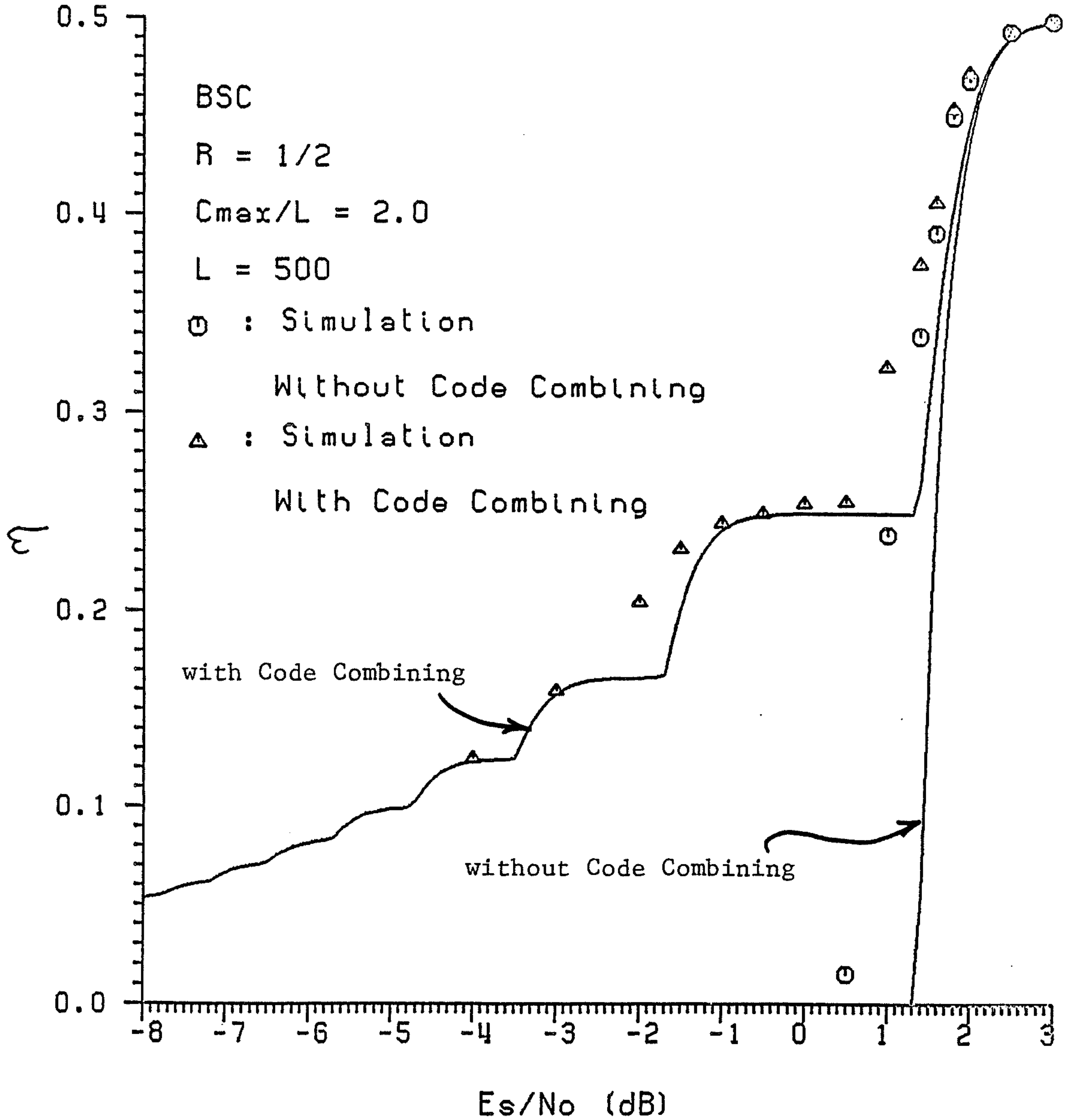

Figure 6 : Numerical and simulation results for the throughput

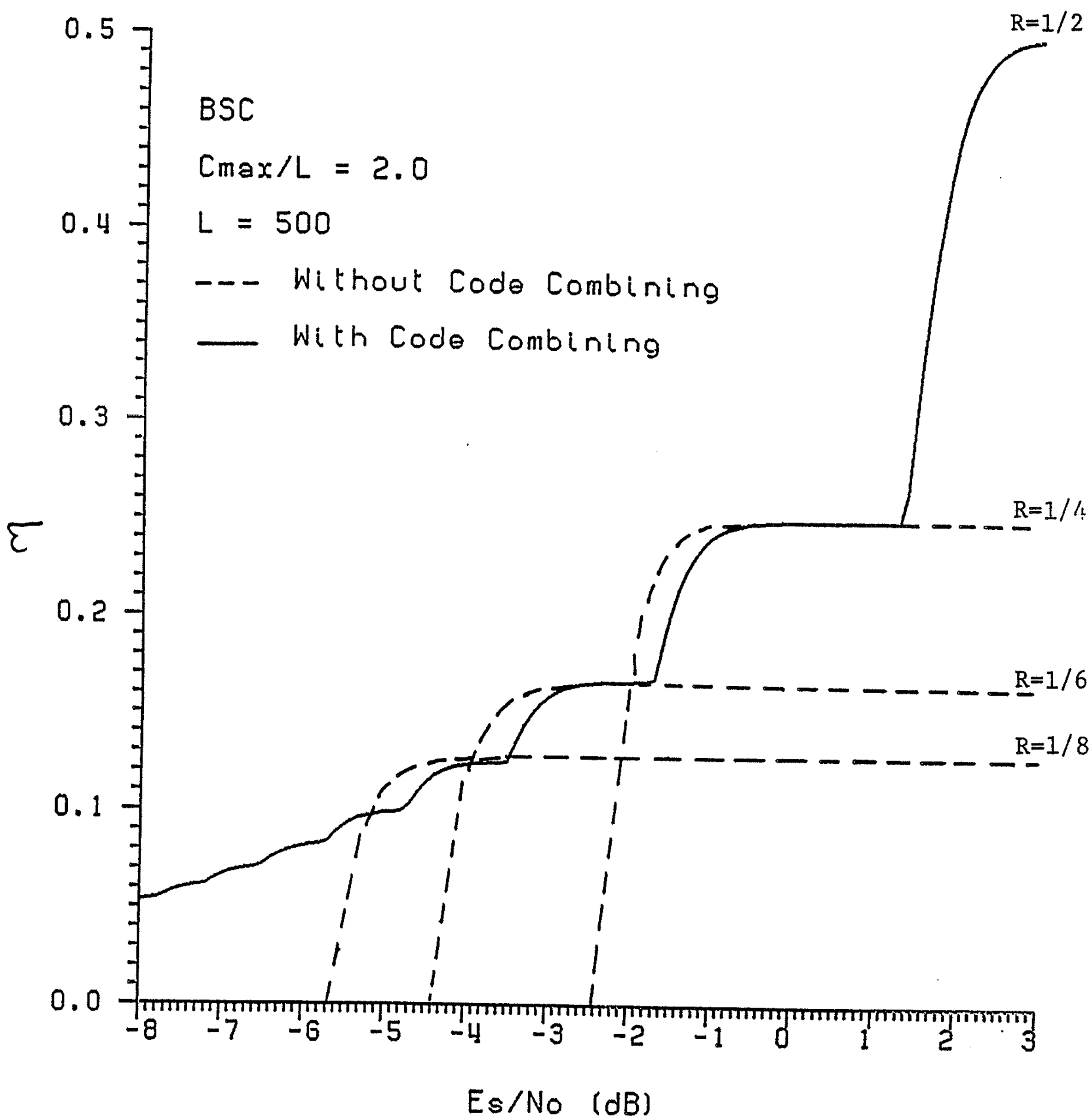

Figure 7 : Throughput comparison of fixed rates 1/4, 1/6, 1/8 decoding without code combining, with a rate 1/2 decoding using code combining

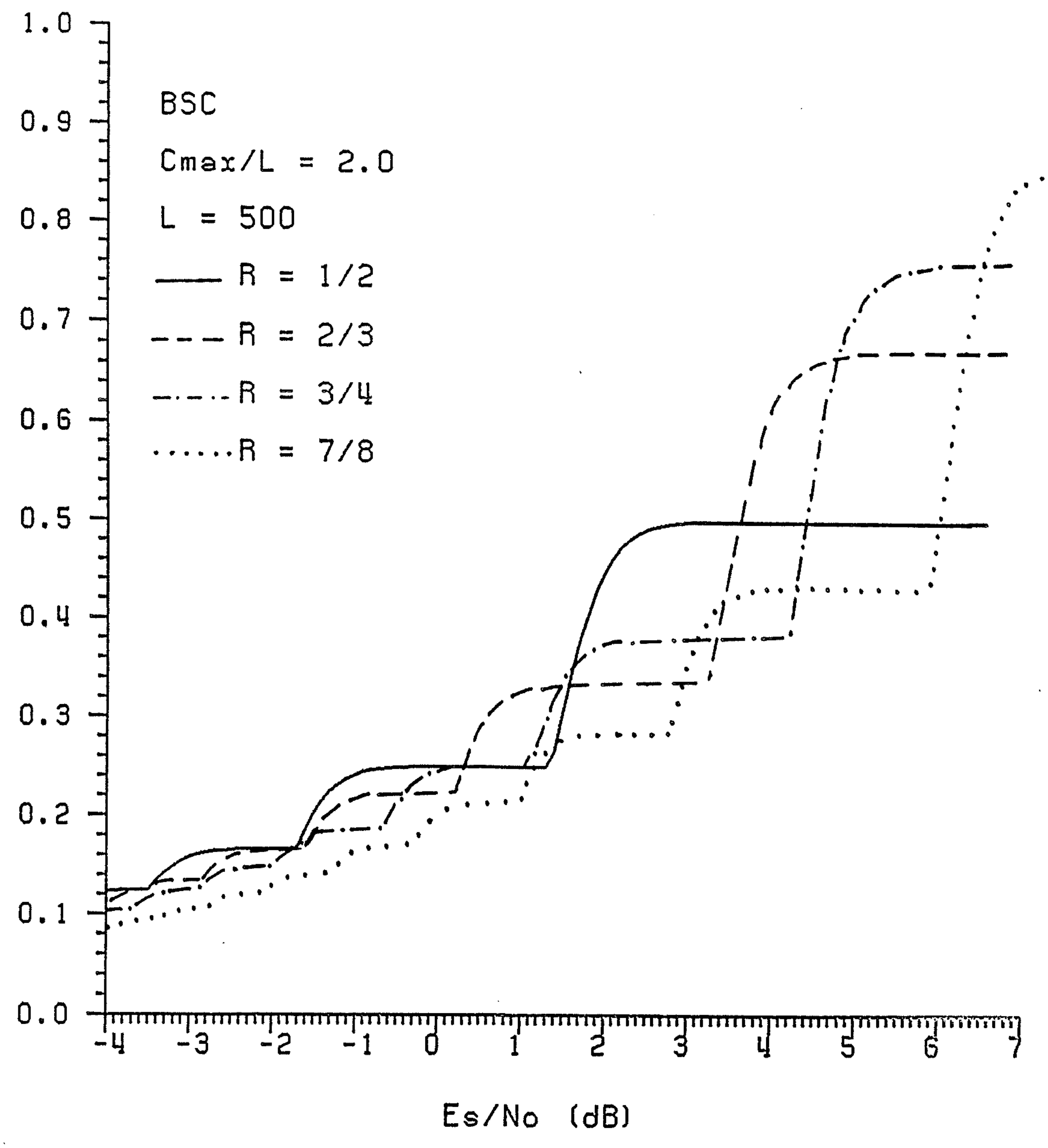

Figure 8 : Throughput for different coding rates with code combining