APPLICATION OF ERROR MODELING AT THE OUTPUT OF MAXIMUM LIKELIHOOD DECODER TO CONCATENATED CODED 16 PSK

K. Fazel, P. Salembier

Laboratoires d'Electronique Philips, 3, avenue Descartes
94451 Limeil-Brévannes Cedex (France)

Abstract

Concatenated coding schemes are widely used in digital communication systems. One interesting combination is based on a convolutional inner code and a block outer code. The purpose of this paper is to study the performance of coded 16 PSK (inner code) concatenated with a Reed-Solomon (RS) code (outer code). Since the performance of the outer code is strongly influenced by the error characteristics of the inner decoder, which is based on the maximum likelihood principle, the errors which result from this type of decoding are modeled. This model is then used to evaluate the performance of different RS codes. Particular attention is paid to the memory effect of the inner Viterbi decoder.

I. Introduction

The combined use of coding and modulation techniques is widely employed in digital communication systems. Indeed, the power efficiency or spectral characteristics of the transmission system can be modified by coding the information on the transmitter side. For example to improve the power efficiency, the redundancy introduced by coding is transmitted by raising the data rate (this is the "classical" error protection approach) or by increasing the size of the signal alphabet (this is the "coded modulation" approach introduced by Ungerboeck [1]). Furthermore, the transmitted signal spectrum can be shaped using techniques such as correlative level coding.

With regard of power efficiency improvement, a number of recent studies [2-3] have shown that a 3 to 4 dB coding gain in signal to noise ratio (SNR) can be achieved with simple codes. However, it is very difficult to get higher coding gain with single codes, because of the drastic increase in decoder complexity. An alternative solution consists of using a concatenated scheme in which the coding process consists of two or more simple codes. One interesting combination is based on a convolutional inner code and a block outer code. Moreover, the inner decoder may follow maximum likelihood (ML) principle (Viterbi decoder) and the outer decoder may use algebraic techniques.

The outer code performance is strongly influenced by the error characteristics at the output of the inner decoder. The latter may occasionally choose a sequence which diverges from the correct one and remerges after l steps. This is an "error-event", containing several consecutive errors. The decoder behavior is such that errors occur in bursts separated by fairly long error-free gaps. This reflects the presence of a memory effect or equivalently the correlation of errors at the decoder output.

From this observation, several questions arise such as : is there a relation between the number of the inner states code and the correlation of error sequences at the output of the corresponding decoder ? Is an interleaver required between the two codes ? What is its optimal depth ? It is possible to take advantage of the presence of this correlation ? Of course, the answer to this last question may influence the code design.

In order to study precisely the combination of inner and outer codes, a very interesting approach consists of modeling the errors at the output of the inner decoder. Indeed, an error model allows to answer to the above mentioned questions, to estimate the outer code performance, and also to regenerate inner error sequences without actually encoding and decoding the signal. As said previously, the inner code creates a memory effect. It therefore cannot be represented by the Binary Symmetric Channel model (BSC). Classical methods of modeling "channel with memory" can be used. They are often based on Markov model consisting of a finite number of states with defined transition probabilities. These probabilities are derived from the knowledge of the error distribution. However, computer simulations used to establish this distribution are not very practical and often require an excessive amount of CPU time. To overcome this problem, a new method of modeling the errors issued from ML (Viterbi) decoders has been introduced [4]. This model is closely related to the decoder error producing mechanism, and the main advantage of this method is its simplicity and the analytical derivation of its parameters (which uses the code "Generating Function" technique).

The purpose of this article is to apply the modeling method to the coded 16 PSK and to study the performance of a concatenated coding scheme involving a convolutional inner code and a block outer code (Reed-Solomon codes). The overall redundancy introduced by the codes is 33 % (Spectral efficiency 2.7 bit/s/Hz), 10 % for the outer code (transmitted at the expense of data rate) and 25 % for the rate 3/4 convolutional code (transmitted by increasing the size of the alphabet from 8 to 16).

This paper is organized as follows : the error modeling method is presented in Section II, techniques used to evaluate the performance of the outer block code is given in Section III, and finally numerical results are given and discussed in Section IV. Particular attention is paid to the memory effect of the inner Viterbi decoder.

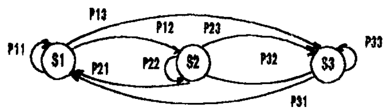
II. Modeling of errors issued from a Viterbi decoder

a) Description of modeling

The model described here is a Markov model with two error-free states (good states) and a single error state (bad state). These three states correspond to the Viterbi error producing mechanism. As it has been said previously, the Viterbi algorithm searches the trellis for the path which is "most likely". The decoder occasionally chooses a wrong path. This is defined as an error event, however all the digits corresponding to a wrong path are not necessarily incorrect. Thus, the first state (S_1) represents the digits belonging to the correct path (of

CH2682-3/89/0000-1528 $1.00 © 1989 IEEE

course it is an error-free state). The two remaining states are devoted to the error events. The first (S_2) corresponds to the correct digits of the wrong paths (error-free state) and the last (S_3) represents the erroneous digits of the wrong paths (error state). This model is illustrated by the following diagram :



$$P = \begin{bmatrix} P_{11} & P_{12} & P_{13} \\ P_{21} & P_{22} & P_{23} \\ P_{31} & P_{32} & P_{33} \end{bmatrix}$$

where :

$$\sum_{i=1} P_{ij} = 1,\ j = 1,\ 2,\ 3$$

where P represents the corresponding transition probability matrix.

The main advantage of this model is that, its transition probabilities (P_{ij}) can be derived analytically, in contrast to classical methods of modeling bursty channels [5]. The analytical derivation of the model parameters is based on the "Generating Function" technique. This function contains in a compact form, all the possibilities of choosing a wrong path instead of a correct one. The statistics needed to compute the transition probabilities of the model and the derivation of these statistics are summerized next.

b) Analytical derivation of the model parameters

The computation of the model transition probabilities requires the knowledge of several parameters [4], the error event probability Pe, the bit error probability Pb, the average length of an error event l, the probability that the first digit of an error event is wrong Pf and the average number of couples (ab) (a, b, ε {0,1}) during an error event N(ab). All these terms can be easily obtained from the code generating function, with appropriate labelling of the code state diagram. The computation of the first four parameters (Pe, Pb, l, Pf) is well known [6] and can be done by labelling the state diagram with scalars. But the derivation of the last parameter N(ab) requires 2x2 matrix labels [4]. After a brief recall of "Generating Function" techniques, the derivation of the model parameters is described in the following. Some specific points of this computation will be illustrated by taking as an example a two state coded 16 PSK [3].

For linear codes, the generating function can be obtained as the transfer function of the code state diagram with transition labels having the following form [6] : LI^jD^d, when j represents the number of errors created if this transition is selected rather than the correct one and d denotes the distance between the wrong and correct transitions. In general, it is assumed that the all zero path is the correct path. The situation is different for arbitrary trellis codes. Indeed, the distance between an incorrect path and a correct path depends on the correct path. To cope with this case, a generalized generating function [2] can be used. It enumerates all possible incorrect paths for all possible correct paths. However its complexity grows exponentially as 2^{2n} where n is the number of code states. To reduce the method complexity, an interesting approach was proposed in [7]. Although, this technique is not suitable for all codes, it is still useful, since it can be applied to many good codes. Furthermore, its complexity is proportional to 2^n rather than 2^{2n}.

To obtain the generating function by this method, a new type of label $LI^jF(D)$ is used, instead of LI^jD^d. F(D) is the so-called "average weight profile" of the transition which generalizes the notion of transition probability D^d. The reader is referred to [7] to see how the average weight profiles can be computed and when this method can be used. Here, a two-state coded 16 PSK scheme is taken as an example. Its state diagram is given in Fig. 1. For each transition, we give the three bit error symbol, the phase of the modulation vector (16-PSK) and its corresponding label : $LI^jF(D)$. $LI^jF(D)$ are given in the table of Fig. 1. The generating function issued from the state diagram is denoted by T (L, I, D) in the following. It is well known [6] that this function is a very efficient method of computing statistics such as the error event probability Pe, the bit error probability Pb and the average length of an error event l :

The Error Event probability :

The error event probability can be derived from the generating function as :

$$Pe = \gamma T(L,I,D)\,|_{L=I=1,\ D=Z}$$

in the case of additive white Gaussian noise :

$$Z = \exp(-\frac{E_s}{4N_0});\quad \gamma = Q((\frac{d_f^2 E_s}{2N_0})^{\frac{1}{2}})\exp(\frac{d_f^2 E_s}{4N_0})$$

where df is the free distance of the code, Es is the symbol energy and No the one-sided noise spectral density.

The Bit Error probability :

Pb can be obtained from the generating function as :

$$Pb = \frac{\gamma}{m}\,\frac{\partial T(L,I,D)}{\partial I}\Big|_{L=I=1,\ D=Z}$$

where m is the number of information bits per transmitted symbol (m = 3 for coded 16 PSK).

The Average Length of error events :

In the same manner l can be expressed as :

$$l = \frac{m\gamma}{Pe}\,\frac{\partial T(L,I,D)}{\partial L}\Big|_{L=I=1,\ D=Z}$$

The Probability that the first bit of an error event is incorrect Pf

This probability can be computed easily, if the state diagram is labelled with F(D) for all transitions except for those corresponding to the begining of an error event and having an incorrect first digit. The label is then IF(D). Pf from this new generating function can be obtained as :

$$P_f = \frac{\gamma}{Pe}\,\frac{\partial T(I,D)}{\partial I}\Big|_{L=I=1,\ D=Z}$$

The last terms to be computed are the average number of error patterns such as (00), (10), (01), (11), where "0" represents a correct bit and "1" an incorrect one, during an error event.

Average number of error patterns during an error event N(ab), (a, b, ε {0,1})

This computation can be done using the generating function. Indeed, it is very easy to count the number of specific couples (ab) inside an error symbol and to consider this number as being the exponent of the dummy variable I. The main difficulty lies in the fact that the transition between the previous and the current error symbol has also to be taken into account. This problem can be solved if the state diagram is labelled with 2x2 matrices instead of scalars.

Let us denote by Ω the set of all possible error symbols. Ω can be split into two subsets Ω_0 and Ω_1, respectively corresponding to the sets of error symbols ending with a correct bit or an erroneous bit. A 2x2 matrix label can be used to distinguish the cases where the current or previous error symbols belong to Ω_0 or Ω_1. The label has the following form :

$$M(I,D) = \begin{bmatrix} M_{\Omega_0\Omega_0}(I,D) & M_{\Omega_0\Omega_1}(I,D) \\ M_{\Omega_1\Omega_0}(I,D) & M_{\Omega_1\Omega_1}(I,D) \end{bmatrix}$$

If the current symbol belongs to Ω_i and the previous one to Ω_j, then M Ω_i Ω_j (I,D) = $I^r F(D)$, where F(D) is the "average weight profile" of the transition and r is the number of pair (ab) in the current error symbol possibly plus one if the pair (ab) occurs in the transition between the error symbols. Of course, the generating function issued from state diagram is itself a 2x2 matrix :

$$T(I,D) = \begin{bmatrix} T_{\Omega_0\Omega_0}(I,D) & T_{\Omega_0\Omega_1}(I,D) \\ T_{\Omega_1\Omega_0}(I,D) & T_{\Omega_1\Omega_1}(I,D) \end{bmatrix}$$

Then the average number of error patterns (ab) is now given by :

$$N(a,b) = \frac{\gamma}{m} \left. \frac{\partial(\sum_{ij} T_{\Omega_i\Omega_j}(I,D))}{\partial I} \right|_{L=I=1,\ D=Z}$$

To illustrate this new procedure, let us take the example of the 2-state coded 16 PSK scheme (Fig. 1). Each transmitted symbol has three information bits, then the two error subsets are defined by Ω_0 = {000, 010, 100, 110} and Ω_1 = {001, 011, 101, 111}. Assume that the error pattern under investigation is (ab) = (11), that the current error symbol is 111 and that the previous one is 101. Since these error symbols belong to Ω_1, the corresponding term in the 2x2 matrix label is MΩ_1 Ω_1(I,D). The pattern (11) can be found twice in the current error symbol 111, and it also occurs in the transition between the two error symbols (101), (111), thus :

$$M_{\Omega_1\Omega_1} = I^3 F(D)$$

The introduction of the matrix labels effectively allows us to count the number of time a specific error pattern occurs during an error event. Let us see now, how the model transition probabilities can be computed with these parameters.

c) Derivation of the model transition probabilities

Transitions emanating from state S_1

P_{12} (resp.P_{13}) is by the definition of the model states, the error event probability knowing that the first digit is an error-free (resp. erroneous) digit. If $P_e < < 1$ (which means that the generating function technique provides a good estimation of Pe) they are given by :

$$P_{12} = \frac{Pe}{3}(1 - Pf);\ P_{13} = \frac{Pe}{3} Pf$$

as: $P_{11} = 1 - P_{12} - P_{13}$; P_{11} is given by: $P_{11} = 1 - \frac{Pe}{3}$

Transitions emanating from state S_2

P22 represents the probability during an error event of an error free bit to follow an error free bit. It is the ratio between the average number of error patterns N(00) and the average number of error-free digits N(0) along a wrong path :

$$P_{22} = P(0/0) = \frac{N(00)}{N(0)} = \frac{N(00)}{l - 3\frac{Pb}{Pe}}$$

Similarly P23 is defined by :

$$P_{23} = P(1/0) = \frac{N(01)}{N(0)} = \frac{N(01)}{l - 3\frac{Pb}{Pe}}$$

and of course : P21 = 1 - P22 - P23.

Transitions emanating from state S_3

For this state, the same procedure as for S_2, can be chosen. It gives the following equations :

$$P_{33} = P(1/1) = \frac{N(11)}{N(1)} = \frac{mPbN(11)}{Pe};$$

$$P_{32} = P(0/1) = \frac{N(10)}{N(1)} = \frac{mPbN(10)}{Pe};\ P_{31} = 1 - P_{33} - P_{32}$$

As can be seen, the approach described above allows us to compute analytically the model transition probabilities. An interesting use of this error model is to evaluate the performance of concatenated coded schemes. The purpose of the next section is to describe how this model can be used to estimate the performance of an outer block code.

III. Performance evaluation of the outer code

One quantity of interest regarding the error performance of block error correction codes is the bit error rate after decoding (BER). Reliable bounds for BER, for two classes of block codes : binary cyclic codes (BCH codes) and non-binary cyclic codes (Reed-Solomon Codes) are given below, and the influence of interleaving is discussed.

a) Binary Cyclic Codes : C (N,K)

Let's denote P(m,N), the probability that exactly m errors occur in a codeword of length N. Suppose that the decoder cannot correct more than t errors, and in the worst case, it does not add more than t errors.

The BER will be defined as :

$$BER = \frac{E(xm)}{N}$$

where E(xm) is the average of the number of errors that occur after decoding. A reliable bound for BER is given by :

$$BER \leq \frac{1}{N} \sum_{i=t+1} (i+t)P(i,N)$$

To compute P(m,N), from the model, we define $f_j(m,N)$, the probability of exactly m transitions into an error state in N steps, given that the initial state was j. Let P_j be the j^{th} state probability, then :

$$P(m,N) = \sum_{j=1}^{3} P_j f_j(m,N)$$

where

$$P_j = \sum_{k=1}^{3} P_k P_{kj}$$

and $f_j(m,N)$ is determined from the following iteration :

$$f_j(m,n) = \sum_{l=1}^{2} p_{jl} f_l(m,n-1) + P_{j3} f_3(m-1,n-1)$$

which can be computed from the following initial conditions :

$$f_i(m,N) = 0 \text{ for } M > N \text{ or } M < 0; \quad f_i(0,0) = 1$$

b) Non Binary Cyclic Codes : (N,K)

The transition probability matrix coefficients that we defined earlier are the transition probabilities between bits, so far a non-binary code ; the model has to be extended [8]. The extended model consists of q + 1 states (q is the number of bit making up the symbol), including one good state and q bad states. The state i(i : 0,1...q) represents a symbol with i errors. Then the probability between symbol states are defined as :

$$P_{Sij} = \frac{P_S(i,j)}{P_{Si}}; \quad i = 0,1..q; \ j = 0,1..q$$

where P_S (i,j) mean that in the error sequence, the states i and j are adjacent, and P_{Si} denotes the i^{th} state probability in the symbol model.

To compute P_{Si} and P_{Sij}, let's define f_{kl} (i,q) as the probability of i transitions leading to an error state for q steps in the bit model, given that, the initial state is k and the final state is l, then this probability can be computed iteratively as follows :

$$f_{kl}(i,q) = \sum_{j=1}^{2} P_{kj} f_{jl}(i,q-1) + P_{3k} f_{k3}(i-1,q-1)$$

with the following initial conditions :

$$f_{kl}(i,q) = 0, \ i < 0, \ q < 0, \ i > q$$

$$f_{kl}(0,0) = 0, \ k \neq l; \ f_{kl}(0,0) = 1, \ k = l$$

then P_{Si} and P_{Sij} can be derived as :

$$P_{Sj} = \sum_{k=1}^{3} P_k [\sum_{r=1}^{3} f_{kr}(j,q)]; \ j = 0,1..q; \ \text{where} \ \sum_{j=0}^{q} P_{Sj} = 1$$

$$P_{Sij} = \frac{\sum_{k=1}^{3} \sum_{l=1}^{3} P_k f_{kl}(i,q) [\sum_{r=1}^{3} f_{lr}(j,q)]}{P_{Sj}}$$

the bit error rate after decoding is now bounded by :

$$BER \leq \sum_{m=t+1}^{N} \sum_{w=m}^{mq} \frac{(w+qt)}{Nq} P(m,w,N)$$

where t is the number of correctable symbol errors and P(m,w,N) denotes the probability that m error symbols and w error bits occur in N symbols. This last term is given by :

$$P(m,w,N) = \sum_{i=0}^{q} P_{Si} f_{Si}(m,w,N)$$

where $f_{Si}(m,w,N)$ is the probability that a block of N symbols has m error symbols and We error bits, given that the initial symbol state is i. The $f_{Si}(m,w,N)$ can be computed as :

$$f_{Si}(m,w,N) = P_{Si0} f_{S0}(m,w,N-1) + \sum_{j=1}^{q} P_{Sij} f_{Sj}(m-1,w-j,N-1)$$

with the following initial conditions :

$$f_{Sj}(m,w,N) = 1, \ m = w = N = 0$$

$$f_{Sj}(m,w,N) = 0, \ m > N, \ w > mq, \ m < 0, \ N < 0, \ w < 0$$

c) Influence of interleaving on the performance evaluation

Interleaving is a way to rearrange the order of a sequence of symbols in a one to one determined manner. This technique is used to combat the effect of burst errors, i.e. to split the error bursts. The distance between two contiguous symbols in original sequence will be n after interleaving and is known as the "interleaving depth". The transition matrix probability (P) of the model (bit model or the extended non-binary model) after interleaving is Q = P^n. This result comes from the property of a Markovien-process [9]. The optimal interleaving depth is determined, when the transition probability from state i to state j becomes independent i.e. all rows of the matrix Q is the same. The expression for the BER after interleaving is exactly the same as we saw before except that the transition probability matrix is now Q.

IV. Performance of concatenated coded-16 PSK

As was stated in the introduction, the purpose of this study is to apply the modeling method to coded 16 PSK and to investigate the performance of the concatenated scheme in the presence of an additive white Gaussian noise (AWGN). The inner code is a 3/4 rate convolutional code and the corresponding decoder is based on the Viterbi algorithm. The following codes have been considered : one with 2 states which provides moderate coding gain over an uncoded 8-PSK (1 dB at a BER = 10^{-11}), and the other one with 16 states which gives higher gain (3.9 dB, same BER) at the expense of an increased complexity. They are optimal codes for 16 PSK and for each complexity [3].

The modeling method described in section II has been applied to these two inner codes. The reliability of this technique is illustrated in Fig. 2 which represents the Error Gap Distribution (EGD). The EGD is defined as the probability of having at least j successive error free bits after an erroneous bit : $P(0^j/1)$. The EGD can be expressed as :

$$P(0^j/1) = \sum_{v=1}^{2} f_v(\lambda)\lambda_v^j$$

where λ_v represent the eigenvalues of the transition probability matrix P, and $f_v(\lambda)$ is a function depends on the state probability, the transition probability matrix and the eigenvectors of the matrix P [10].

Fig. 2 presents the analytical results (in solid line) for the 2 and 16 state codes, as well as simulation results (dotted line) which were obtained by actually encoding and decoding a very long pseudo-random sequence. It can be seen that the analytical and simulation results are in very good agreement. All the results are given for a bit error rate (BER) of 3.610^{-4}. It should be noted that the use of the generating function technique and thus modeling technique give accurate results for BER lower than 10^{-3}. (This is due to the use of the union bound which is not tight for every high BER).

The EGD also reflects the correlation of the error sequence. In the case of a memoryless error-sequence, the EGD is simply given by :

$$P(0^j/1) = (1-p)^j \; ; \quad \text{where } p = \text{BER}$$

This exponential curve is given in fig. 2 for a BER of 3.610^{-4}. It can be seen that the EGD of the two-state inner code is not a simple exponential function. This is an indication of a memory effect. For the 16-state inner code, the EGD behavior is even further from the memoryless case. It can be conclued that the error sequence is more effectively correlated at the output of the 16 state decoder than at the output of the 2-state decoder. Of course, this result is consistent with common sense which suggests that the 16-state code has a longer memory than the 2-state code.

The following figures (Fig. 2-Fig. 7) represent the performances of concatenated coding schemes. The inner code is a 2-or 16-state convolutional code. As the error sequence created by Viterbi decoders are correlated, an outer code known to be efficient for bursty errors sequences was chosen, namely a Reed-Solomon (RS) code.

The redundancy of the outer code was fixed to 10 %. Finally the sequence was interleaved between the two codes. The influence of the interleaving depth is represented in Fig. 3 and Fig. 4. The RS code is based on a codeword of length 63 (6 bits per symbol) and can correct up to three error symbols. The interleaving works with symbols (the correlation of error bits in the inside of symbols is conserved).

Fig. 3 gives the BER as a function of the signal to noise ratio (Eb/No) for various interleaving depths, and in the case of 2-state inner code it can be seen that at a BER of 10^{-8}, a gain of 2 dB in noise power can be obtained by interleaving. Moreover, optimal results are obtained with a very simple interleaving of depth 5 symbols. In the case of 16-state inner code (Fig. 4) a higher interleaving depth is required to get optimal results (at least 40), but the gain in noise power is much higher (3 dB). These results show that interleaving is desirable between the two codes, and that the depth depends heavily upon the inner code complexity.

The influence of the outer code is illustrated in Fig. 5 and Fig. 6. In each case the BER at the output of the two inner decoders (2 and 16 state) and the outer decoder as a function of the signal to noise ratio (Eb/No) is given. Fig. 5 (resp. Fig. 6) refers to a RS code of length 63 (resp. 127) which can correct 3 errors (resp. 6 errors) - in both cases, optimal interleaving depth was chosen. In Fig. 7, the BER_o at the output of the RS decoder is represented as a function of the input BER_i.

These curves clearly show the advantages of the RS codes. The coding gains (at a BER = 10^{-11}) over an uncoded 8-PSK (Fig. 5-6), provided by the RS (63, 57) code (resp. RS(127, 115) code) are respectively 4.6 dB (resp. 5.7 dB) and 6.7 dB (resp. 7.3 dB) for 2-state and 16-state inner codes. Moreover, the influence of the error sequence correlation at the input of the RS code can be examined. Indeed, the 16-state inner code produces error sequences with a higher correlation than the 2-state code. This higher correlation allows us to decrease the output BER (Fig. 7) by a factor of 3 (resp. 10) with a RS (63,57) code (resp. with RS (127,115) code) for the same input BER. Those results seem to indicate that higher efficiency is obtained by using an inner code which creates bursty error sequences and an outer code which is suitable for handling bursty errors.

Conclusion

In this paper, the problem of modeling the errors at the output of a maximum likehood decoder has been considered. The model presented in closely related to the decoder error producing mechanism. Its main advantage is the analytical derivation of its parameters, based on the code "Generating Function".

The modeling method has been applied to concatenated 16 PSK schemes. The inner codes were rate 3/4 convolutional codes with 2-or 16-states, and the outer codes were Reed-Solomon codes (RS (63, 57) or RS (127, 115)). Finally symbol interleaving was used between two codes. The use of the model made the study of the influence of the interleaving depth simple, and to determine its optimal depth. It also allows the estimation of the global performance of concatenated schemes, and to evaluate the influence of the error sequence correlation i.e. the influence of the inner code.

References

[1] G. Ungerboeck, "Channel coding with multileval/phase signals", IEEE Trans. on IT, vol. 28, pp. 55-66, June 1982.

[2] E. Biglieri, "High level modulation and coding for non-linear satellite channel", IEEE Trans. on Com., vol. 32, pp. 616-628, May 1984.

[3] S.G. Wilson et al., "Rate 3/4 convolutional coding of 16 PSK : code design and performance study", IEEE Trans. on Com., vol. 32, pp. 1308-1314, Dec. 1984.

[4] K. Fazel, P. Salembier, "Modeling of errors issued from maximum-likelihood decoding", GRETSI 89, pp. 271-274, Juan-les-Pins, France.

[5] L. Kanal et al., "Models for channels with memory and their application to error control", Proc. of IEEE, vol. 66, pp. 724-743, July 1978.

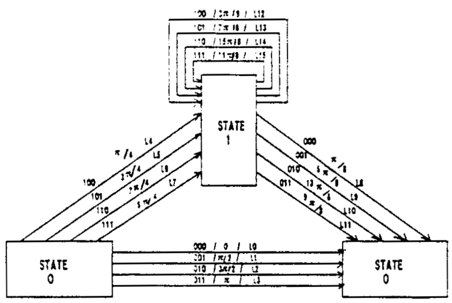
[6] Ommura-Viterbi, "Digital Communication and Coding", McGraw Hill, N.Y., 1979.

[7] E. Zehavi, J.K. Wold, "On the performance evaluation of trellis codes", IEEE Trans. on IT, vol. 33, pp. 196-202, March 1987.

[8] A.I. Drukarev, "Performance of error-correcting codes on channels with memory", IEEE Trans. on Com., vol. 34, pp. 513-521, June 1986.

[9] S. Tsai and P. Schnied, "Interleaving and error-burst distribution", IEEE Trans. on Com., vol. 2, pp. 291-296, June 1972.

[10] B. Fritchman, "A binary channel characterization using partitioned Markov chain", IEEE Trans. on IT, vol. 13, pp. 221-227, April 1967.



Li	Transition label	Li	Transition label
L0	U^0D^0	L8	U^0D^{d1}
L1	U^1D^{d4}	L9	$U^2.5(D^{d3}+D^{d5})$
L2	U^1D^{d4}	L10	$U^2.5(D^{d3}+D^{d5})$
L3	U^2D^{d8}	L11	U^2D^{d7}
L4	U^1D^{d2}	L12	$U^1.5(D^{d1}+D^{d3})$
L5	$U^2.5(D^{d2}+D^{d6})$	L13	$U^1.25(D^{d1}+D^{d3}+D^{d5}+D^{d7})$
L6	$U^2.5(D^{d2}+D^{d6})$	L14	$U^1.25(D^{d1}+D^{d3}+D^{d5}+D^{d7})$
L7	U^3D^{d6}	L15	$U^1.5(D^{d5}+D^{d7})$

$$di = (2\sin(\frac{i\pi}{16}))^2$$

$$D = \exp(\frac{-Es}{4N0})$$

Fig.1 2 state encoder State Diagram

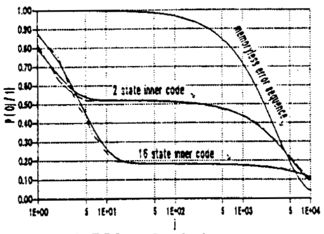


Fig.2 EGD for 2 and 16 state inner codes(BER = 3.6E-4)

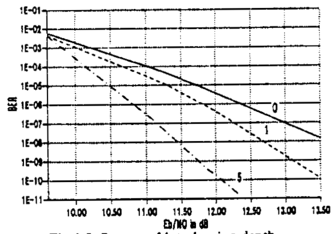


Fig.3 Influence of interleaving depth (2 state inner code + RS(63,57))

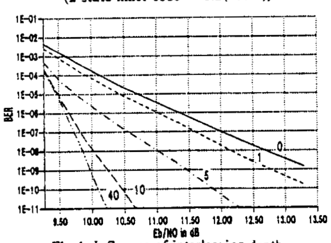


Fig.4 Influence of interleaving depth (16 state inner code + RS(63,57)

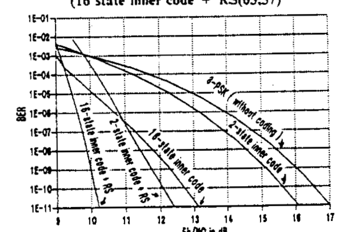


Fig.5 Performances of concatenated scheme with RS(63,57)

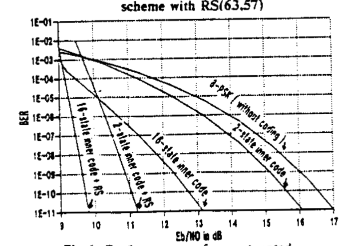


Fig.6 Performances of concatenated scheme with RS(127,115)

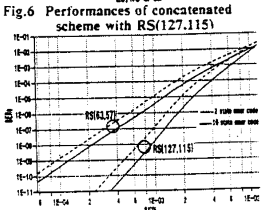


Fig.7 BER at the output of the concatenated scheme