# Exact Calculation of the Union Bound on Performance of Trellis Coded Modulation in Fading Channels

by

James K. Cavers
School of Engineering Science
Simon Fraser University
Burnaby, B.C., Canada

Jae Hyung Kim
Department of Control and
Instrumentation Engineering
Changwon National University
Changwon, S. Korea

Paul K.M. Ho
School of Engineering Science
Simon Fraser University
Burnaby, B.C., Canada

## ABSTRACT

This paper presents a method for exact calculation of the union bound on error performance of trellis coded modulation in fading channels. Previous analyses have approximated the error expression to allow use of the transfer function, or restricted their attention to the most common error events, or both. In contrast, we combine the transfer function, which counts all error events, with exact calculation of event probabilities, and produce an exact expression for the union bound. Although computationally intensive, the technique produces the tightest bound yet published. In addition, it is very general and is applicable to coherent, differential and pilot-based detection, and to PSK and QAM types of constellations.

Corresponding Author:
Dr. Paul K.M. Ho
School of Engineering Science
Simon Fraser University
Burnaby, B.C., V5A 1S6
Canada

tel: (604) 291-3822 fax: (604) 291-4951 email: paul@cs.sfu.ca

March, 1996

This work was supported by the National Sciences and Engineering Research Council of Canada.

# I. INTRODUCTION

Trellis coded modulation (TCM) is a popular method of error control for fading channels. Its principal advantages include a straightforward implementation and the incorporation of time diversity when the channel symbols are interleaved. Accurate evaluation of the error performance of a specific code is clearly important, but it has proved challenging to obtain analytically, and many researchers have simply resorted to simulation.

There have been several published analyses of the error performance of TCM in fading; for example [1]-[5]. Their common starting point is the union bound, that is, the sum of the probabilities of all error events that begin at an arbitrary point in time. However, the true union bound is difficult to calculate, and most authors employ one or both of two principal approximations. In one approach, for example [1], [4] and [5], the pairwise error probability is approximated, usually by a Chernoff bound, to put it in a product form, and the sum over all pairwise events is achieved by use of the transfer function of the underlying convolutional code. In the second, [2]-[3], the pairwise event probabilities are calculated exactly, but only for a finite number of dominant events.

In the present paper, we describe a way to combine the transfer function with a previously published technique [3] for exact calculation of the pairwise error events. In this way, we calculate the sum of error probabilities for ***all*** error events, and generate an exact expression for the true union bound (TUB). It is unlikely that a closer bound will be obtained. More precisely, we generate a Laplace transform which specifies the TUB. For Rayleigh fading channels, inversion of the transform is readily achieved symbolically by residues or numerically by explicit integration. The method is quite general, and can handle multilevel constellations, imperfect channel state information, such as differential or pilot aided detection, and diversity reception.

For demonstration purposes, we concentrate on M-ary phase shift keying (MPSK) modulation. Numerical results for the TUB will be provided for (1) a simple rate 1/2, two-state QPSK code, (2) a rate 1/2, four-state BPSK code, and (3) Ungerboek's four- and eight-state, rate 2/3 8PSK codes. Comparisons will be made with the tight bound in [5] in the case of perfect coherent detection, and with [4] in the case of differential detection.

## II. SYSTEM MODEL

### *A. Model Overview*

Fig. 1 shows the link model. Data bits at symbol time $k$ are accepted as a binary $m$-tuple $\boldsymbol{a}(k)$ and are convolutionally encoded to generate an $(m+1)$-tuple $\boldsymbol{c}(k)$ which selects a complex scalar $x(k)$ in the symbol mapper using the mapping function $x=f(\boldsymbol{c})$. The scalar symbols are given perfect (i.e. effectively infinite) interleaving to produce the sequence $x'(k)$, and the interleaved sequence is applied to a unit energy pulse shaping filter, which we assume to have a root Nyquist characteristic. The complex envelope of the resulting transmitted signal is $s(t)$.

The flat fading process in the channel multiples the transmitted signal by a complex Gaussian gain $g(t)$ and the receiver's front end adds complex Gaussian noise with a flat power spectrum of value $N_0$. The receiver detection process consists of a filter matched to the transmitted pulse, and normalized by $N_0^{1/2}$. Its output is sampled at the symbol rate to produce the sequence r'(k) and the estimate, where the prime (') indicates quantities before deinterleaving. The receiver also produces an estimate $v'(k)$ of the channel gain, typically by means of a pilot tone or pilot symbols, or by use of the matched filter output from one or more previous symbols as a reference, as in differential detection. Both the filter outputs and the channel estimate are deinterleaved to produce the sequences $r(k)$ and $v(k)$ that enter the Viterbi decoder for trellis decoding. Because of the assumed infinite interleaving depth, successive samples of $r(k)$ are independent when conditioned on the transmitted signal, as are successive samples of $v(k)$.

Assuming the fading to be slow compared to the symbol rate, we can write the samples at the output of the deinterleaver as

$$r(k) = u(k)\,x(k) + n(k) \tag{1}$$

where the normalized gain is

$$u(k) = \frac{g(k)}{\sqrt{N_o}} \tag{2}$$

and $n(k)$ is the $k$th noise sample.

### *B. Power Normalization and Correlations*

We define the constellation so that $E[|x(k)|^2] = 1$, where E[ ] denotes expectation. The channel gain is Gaussian with variance $\sigma_g^2 = \frac{1}{2}\,E[|g(k)|^2]$, so that the normalized gain is Gaussian with variance

$$\sigma_u^2 = \frac{\sigma_g^2}{N_o} = \Gamma_s, \tag{3}$$

where $\Gamma_s$ is the signal to noise ratio per symbol, and the second equality follows from the fact that the noise variance $\sigma_n^2 = 1$. We also have the variance and conditional variance of $r(k)$ as

$$\sigma_r^2 = \Gamma_s + 1 \tag{4}$$

and

$$\sigma_{r|c}^2 = \Gamma_s |x(k)|^2 + 1 . \tag{5}$$

For effective operation, the channel estimate $v(k)$ must be highly correlated with $u(k)$. We define the covariance and corresponding correlation coefficient as

$$\sigma_{uv}^2 = \frac{1}{2} E[u(k)\, v^*(k)], \tag{6}$$

$$\mu = \frac{\sigma_{uv}^2}{\sigma_u \sigma_v} . \tag{7}$$

For perfect channel state information (CSI), $\mu$=1. Finally, we have the conditional covariance between $r(k)$ and $v(k)$ as

$$\sigma_{rv|c}^2 = \frac{1}{2} E[r(k) v^*(k) | x(k)] = \sigma_{uv}^2 x(k) . \tag{8}$$

*C. TCM Decoder*

Given received sequences $\boldsymbol{R}$ = (..., $r(k)$, $r(k+1)$,...) and $\boldsymbol{V}$ = (..., $v(k)$, $v(k+1)$,...), the optimum decoder maximizes the a posteriori probability $p(\boldsymbol{C} \,|\, \boldsymbol{R},\boldsymbol{V})$, where $\boldsymbol{C}$ is taken from the set of code words. Equivalently, with equiprobable code words, it minimizes the sum over all instants $k$ of the one-step metrics

$$M(r(k), v(k), c(k)) = -\ln(p(r(k) | x(k), v(k)) \tag{9}$$

since the symbol $x(k)$ is determined uniquely by the code word fragment $c(k)$ through the symbol mapping $x = f(c)$. Now, given $v(k)$, the normalized gain $u(k)$ is Gaussian with mean value $\beta v(k)$ and estimation error variance $\sigma_e^2$, where

$$\beta = \mu \frac{\sigma_u}{\sigma_v}, \tag{10}$$

$$\sigma_e^2 = (1 - |\mu|^2)\,\Gamma_s . \tag{11}$$

The received sample $r(k)$ is conditionally Gaussian with mean $\beta v(k)x(k)$ and variance

$$\sigma^2_{r|cv}[c(k)] = |x(k)|^2\,\sigma_e^2 + 1 . \tag{12}$$

The one-step metric for the optimum decoder can now be written as

$$M(r(k), v(k), c(k)) = \frac{|r(k) - \beta\, v(k)\, x(k)|^2}{2\sigma^2_{r|cv}(c(k))} + \ln\left(\sigma^2_{r|cv}(c(k))\right) \tag{13}$$

and can easily be calculated by a Viterbi algorithm sequence decoder.

## III. PAIRWISE ERROR PROBABILITY

### *A. Basic Expression*

In this section, we summarize the characteristic function method [3] for calculationg the pairwise error event probability. For an error event of length $N_e$, we have the true code word $\boldsymbol{C} = (c(1), ..., c(N_e))$ and an erroneous code word $\hat{\boldsymbol{C}} = (\hat{c}(1), ..., \hat{c}(N_e))$, with corresponding symbol sequences $(x(1), ..., x(N_e))$ and $(\hat{x}(1), ..., \hat{x}(N_e))$. An error is made if the metric of $\hat{\boldsymbol{C}}$ is less than that of $\boldsymbol{C}$, or equivalently, if the difference of metrics is negative; that is,

$$D = \sum_{k=1}^{N_e} d(r(k), v(k), c(k), \hat{c}(k)) < 0, \tag{14}$$

where the one-step metric difference $d(r,v,c,\hat{c})=M(r,v,\hat{c})-M(r,v,c)$. Substituting (13) and rearranging, we can rewrite the one-step difference as

$$d(r,v,c,\hat{c}) = A|r|^2 + B|v|^2 + 2Re[Crv^*] - \delta, \tag{15}$$

where $A$, $B$, $C$ and $\delta$ depend on the code word fragments $c$ and $\hat{c}$, but not on $r$ and $v$. In particular, the offset $\delta$ is given by

$$\delta(c,\hat{c}) = \ln\left(\frac{\sigma^2_{r|cv}(c)}{\sigma^2_{r|cv}(\hat{c})}\right). \tag{16}$$

The metric difference $d$ in (15) is a random variable in which the first three terms are a quadratic form in zero mean complex Gaussian random variables and the last term is a constant offset. Its characteristic function (the two-sided Laplace transform of its probability density function) is well known to be [6,7]

$$\Phi_d(s,c,\hat{c}) = \frac{p_L(c,\hat{c})\,p_R(c,\hat{c})}{(s-p_L(c,\hat{c}))(s-p_R(c,\hat{c}))} e^{s\delta(c,\hat{c})}. \tag{17}$$

The left plane (LP) and right plane (RP) poles $p_L$ and $p_R$ can be written as [3]

$$\begin{bmatrix} p_L(c,\hat{c}) \\ p_R(c,\hat{c}) \end{bmatrix} = w(c,\hat{c}) \mp \sqrt{w(c,\hat{c})^2 - p_L(c,\hat{c})p_R(c,\hat{c})}, \tag{18}$$

where the product of the poles is

$$p_L(c,\hat{c})\,p_R(c,\hat{c}) = -\frac{1+|\hat{x}|^2(1-|\mu|^2)\Gamma_s}{|\mu|^2|\hat{x}-x|^2\Gamma_s} \tag{19}$$

and half the sum of the poles is

$$w(c,\hat{c}) = \frac{1}{2} - \frac{(1-|\mu|^2)(|\hat{x}|^2-|x|^2)}{2|\mu|^2|\hat{x}-x|^2}. \tag{20}$$

For any choice of parameters, $p_L<0$ and $p_R\geq 1$.

Because of the assumed independence resulting from ideal interleaving, the characteristic function of the metric difference $D$ of the error event is

$$\Phi_D(s,C,\hat{C}) = \prod_{k=1}^{N_s} \Phi_d(s,c(k),\hat{c}(k)), \tag{21}$$

where the region of convergence lies between the rightmost LP pole and the leftmost RP pole. The characteristic function can be inverted using Laplace transform methods to give the pdf, which can in turn be integrated to produce the probability that $D$ is negative, that is, the error event probability. It is more convenient, though, to represent the integration with division by $s$ in the transform domain, and define the functional

$$\mathrm{N}[F(s)] = \frac{1}{2\pi j}\int_{\sigma-j\infty}^{\sigma+j\infty}\frac{1}{s}F(s)\,ds \tag{22}$$

for any $F(s)$, where $\sigma$ is chosen to place the integration line in a defined region of convergence (ROC). The error event probability is then given exactly by

$$P_e(C,\hat{C}) = \mathrm{N}\left[\Phi_D(s,C,\hat{C})\right] \tag{23}$$

where the ROC is defined by $0 < \sigma <$ the minimum RP pole.

From the residue theorem, the line integral (22) can be expressed as

$$P_e(C,\hat{C}) = \begin{cases} -\sum\limits_{RP\ poles} Res\left[\Phi_D(s,C,\hat{C})/s\right], & \Delta(C,\hat{C})\geq 0 \\ \sum\limits_{\substack{LP\ poles,\\ s=0}} Res\left[\Phi_D(s,C,\hat{C})/s\right], & \Delta(C,\hat{C})<0 \end{cases} \tag{24}$$

where

$$\Delta(C,\hat{C}) = \sum_{k=1}^{K}\delta(c(k),\hat{c}(k)). \tag{25}$$

*B. Euclidean Metric*

The offset $\Delta(C,\hat{C})$ complicates the expressions in Section IIIA. However, there are at least three important cases in which it can be omitted. The first two are perfect CSI, in which $\sigma_e^2=0$, or the use of constellations in which all $|x(k)|=1$, such as MPSK. Both cases result in all $\sigma_{rlcv}^2$ values being equal, so that all $\delta$ values equal zero and the metric becomes the simple Euclidean metric, given by the numerator of the first term in (13). The third case is use of the Euclidean metric, even when the optimum metric is correct, because it is simpler and there is little cost to doing so. In fact, the optimum metric is useful only when the estimation error term (11) dominates the conditional variance (12); that is, only in the error floor (irreducible error rate) region of multiamplitude constellations. Most system designers avoid this region, if possible, which further reduces its practical importance.

The poles of the characteristic function $\Phi_d(s,c,\hat{c})$ (17) are modified in the case of the Euclidean metric to have the product

$$p_L(c,\hat{c})\,p_R(c,\hat{c}) = \frac{-\left(1+\left(1-|\mu|^2\right)\Gamma_s\right)^2}{|\mu|^2\,\Gamma_s\,|x-\hat{x}|^2\left(1+\left(1-|\mu|^2\right)|x|^2\,\Gamma_s\right)} \tag{26}$$

and half the sum

$$w(c,\hat{c}) = \frac{1+\left(1-|\mu|^2\right)\Gamma_s}{2\left(1+\left(1-|\mu|^2\right)|x|^2\,\Gamma_s\right)} \tag{27}$$

and the exponential factor vanishes, because $\delta=0$.

### *C. Diversity Reception*

If the receiver has access to more than one received signal $R_1$, ..., $R_L$, where the $L$ fading and noise processes are independent, then the metric it feeds to the Viterbi algorithm is the sum of the individual channel metrics (13). Because the metrics are independent when conditioned on the transmitted signal, the characteristic function of the metric difference D of the error event is the product of the individual characteristic functions. If the channels have the same statistics, then (21) becomes

$$\Phi_D(s,C,\hat{C}) = \prod_{k=1}^{N_e} \Phi_d^L(s,c(k),\hat{c}(k))\,. \tag{28}$$

All poles acquire additional multiplicity $L$ and the offset $\Delta$ is scaled by a factor $L$.

## IV. THE TRANSFER FUNCTION BOUND

In this section, we combine the characteristic function method for exact evaluation of the pairwise error event with the transfer function method for enumerating all error events. The derivation parallels that in [8] for the additive white Gaussian noise channel.

### *A. The Trellis Encoder*

Fig. 2 shows an expanded view of the trellis encoder of Fig. 1. At each time step, a symbol from an alphabet of size $M'=2^m$ enters the encoder as the binary $m$-tuple $\boldsymbol{a}(k)$ and the coder selects a complex channel symbol $x(k)$ from a constellation of size $M=2^{m+1}$. Within the encoder, $\tilde{m}$ of the $m$ source bits are convolutionally encoded by a linear code to generate $\tilde{m}+1$ bits which select a subconstellation in the signal mapper. The remaining $m-\tilde{m}$ source bits select a symbol from within the subconstellation. The $(m+1)$-tuple of bits entering the symbol mapper is the code fragment $\boldsymbol{c}(k)$, and the overall mapping function is denoted $x=f(\boldsymbol{c})$.

The convolutional encoder has $N$ states $\sigma_0,\ldots, \sigma_{N-1}$, and can be represented by a trellis diagram, as shown in Fig. 3a, in which transitions are labelled with the input and output bits $\boldsymbol{a}|\boldsymbol{c}$. There are $2^{\tilde{m}}$ transitions into and out of each state, and the $m-\tilde{m}$ uncoded source bits produce parallel transitions within each one.

### *B. The Error Word Transform*

An assessment of the error probability requires a summation over all possible true code words $\boldsymbol{C}$ and all possible erroneous words $\hat{\boldsymbol{C}}$ for each one. A convenient way to organize the computation is to focus on ethe sequence of encoder states to define the true code word, and on the error word $\boldsymbol{E}=\boldsymbol{C}\oplus\hat{\boldsymbol{C}}$, where $\oplus$ denotes the bitwise exclusive or. Accordingly, we define the $N$x$N$ *error transform matrix* $\boldsymbol{G}(s,\boldsymbol{e})$ with $i,j$ component given by

$$[\boldsymbol{G}(s,\boldsymbol{e})]_{i,j} = \frac{1}{M'}\sum_{\boldsymbol{c}_{i\to j}} \Phi_d(s,\boldsymbol{c}_{i\to j},\boldsymbol{c}_{i\to j}\oplus\boldsymbol{e}) \tag{29}$$

where $\boldsymbol{c}_{i\to j}$ denotes the codeword fragment associated with the transition from $\sigma_i$ to $\sigma_j$, $\boldsymbol{e}$ is the fragment of the error word and the sum is over parallel transitions. In the case of diversity reception, $\Phi_D$ should be replaced by its $L^{\text{th}}$ power. Applying the functional (23), (23) gives the conditional joint probability

$$Pr[\boldsymbol{e},\sigma_j|\,\sigma_i] = \mathbf{N}\left[[\boldsymbol{G}(s,\boldsymbol{e})]_{i,j}\right] \tag{30}$$

that $\boldsymbol{e}$ has a better (lower) metric than $\boldsymbol{0}$ and that the next state is $\sigma_j$. For an error word $\boldsymbol{E}=(\boldsymbol{e}_1,\ldots,\boldsymbol{e}_K)$ we obtain the compound error transform matrix by multiplication as

$$G(s,E) = \prod_{k=1}^{K} G(s,e_k) . \tag{31}$$

The functional **N** applied to the $i,j$ element of $G(s,E)$ produces the probability that the error word $E$ has a lower metric than $0$ and the last state of the true state sequence is $\sigma_j$ , given that the initial state is $\sigma_i$.

We can calculate an *error word transform* for $E$ by

$$g(s,E) = \frac{1}{N} \mathbf{1}^T G(s,E) \mathbf{1} \tag{32}$$

where **1** is a length-$N$ column vector. The $N$ states are assumed equiprobable. This is the characteristic function of the metric difference for error word $E$, averaged over all possible correct state sequences, so the pairwise error probability of $E$ is given by $\mathbf{N}[g(s,E)]$.

### *C. The Error State Diagram*

We obtain the overall union bound on probability of an error by summing the pairwise error probabilities, as obtained above, over all error words. If the probability of individual error words decreases exponentially with length more quickly than the exponential growth of number of code words, we obtain a finite sum. At very low SNR or very poor CSI, however, the sum may diverge. This is a general property of the true union bound, irrespective of calculation or approximation method.

Because the allowable error words are constrained by the convolutional code, a straightforward way to enumerate them is the error state diagram. Since the convolutional code is assumed to be linear, the error state diagram is the same as the state diagram for the code itself, except for the fact that the all-zero state (i.e. the error-free state) is split into a source and a sink. Fig. 3c shows the error state diagram corresponding to the trellis of Fig. 3a, with states denoted as $\varepsilon_k$ to distinguish them from code states $\sigma_k$.

We associate with the transition from state $\varepsilon_m$ to $\varepsilon_n$ the $N$x$N$ *label matrix* $F(s,I,J,e)$, defined as

$$F(s,I,J,e_{m\to n}) = I^b J \sum_{e_{m\to n}} G(s,e) , \tag{33}$$

where $I$ and $J$ are supplementary variables, $b$ is the number of source bits in error as a result of the transition, and the sum is taken over parallel transitions. A *matrix transfer function* $T(s,I,J)$ is obtained by calculating the transfer function of the resulting graph, and the *error transform* is just the scalar

$$t(s,I,J) = \frac{1}{N}\mathbf{1}^T T(s,I,J)\mathbf{1}. \tag{34}$$

Note that $t(s,I,J)$ is not a true characteristic function, and does not necessarily equal unity at $s=0$, since the pairwise probabilities do not have to sum to unity.

Implicit in calculation of the transfer function is the assumption of convergence of the various geometric series introduced by feedback loops in the state diagram. This requirement introduces a number of subtle convergence issues that are best introduced by example, as in Sections V and VI.

*D. Error Probabilities*

The error transform determines the error probabilities. The functional **N** is linear, so we can apply it to the error transform $t(s,I,J)$, and obtain a number of useful quantities. The probability of an error event is given by

$$P_e = \mathbf{N}[t(s,I,J)]_{I=1,J=1}. \tag{35}$$

More precisely, this is the true union bound (TUB) on the error performance of the code, since it is the sum over all possible error events of the exact values of the pairwise probability of those events.

By differentiation, the TUB on the probability of a bit error is given by

$$P_b = \frac{1}{m}\mathbf{N}\left[\frac{\partial}{\partial I}t(s,I,J)\right]_{I=1,J=1}, \tag{36}$$

and the TUB on the probability of a symbol error is

$$P_{sym} = \mathbf{N}\left[\frac{\partial}{\partial J}t(s,I,J)\right]_{I=1,J=1}. \tag{37}$$

Further, the average length of an error event is simply $N_{av}=P_{sym}/P_e$.

## V. MPSK OR PERFECT CSI

In this section, we illustrate our method of calculating the true union bound (TUB) on performance with examples of two codes, and demonstrate that considerable simplicity is gained with either of two special conditions - MPSK codes or perfect CSI. Section VI deals with more general codes and conditions.

### *A. Two-State QPSK Code*

As a simple example of the method, we calculate the true union bound on performance of a two-state 4PSK code. With $m$=1, there are no parallel transitions. Since it is a PSK code, all the points in the signal constellation (Fig.3b) have unit magnitude. Consequently, the pole locations (18) of the elementary characteristic functions (17) simplify to

$$\begin{Bmatrix} p_L(c,\hat{c}) \\ p_R(c,\hat{c}) \end{Bmatrix} = \frac{1}{2} \mp \sqrt{\frac{1}{4} + \frac{P_s}{|d|^2}} \tag{38}$$

where $d=x-\hat{x}$ and

$$P_s = \frac{1 + (1-|\mu|^2)\Gamma_s}{|\mu|^2 \Gamma_s} \tag{39}$$

The constellation points enter only as their squared separations and, from the signal mapping, the separations are determined by the error word fragment $e$. Thus all elements of an error transform matrix have the same value, giving

$$G(s,10) = \Phi_4(s)\frac{1}{2}\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \tag{40}$$

$$G(s,01) = G(s,11) = \Phi_2(s)\frac{1}{2}\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \tag{41}$$

where $\Phi_2(s)$ and $\Phi_4(s)$ are the elementary characteristic functions (17) with poles at

$$\begin{Bmatrix} p_{2L} \\ p_{2R} \end{Bmatrix} = \frac{1}{2}\left(1 \mp \sqrt{1+2P_s}\right) \tag{42}$$

$$\begin{Bmatrix} p_{4L} \\ p_{4R} \end{Bmatrix} = \frac{1}{2}\left(1 \mp \sqrt{1+P_s}\right) \tag{43}$$

respectively.

From the error state graph, we obtain the matrix transfer function

$$\begin{aligned} T(s,I,J) &= IJG(s,10)\left(I + IJG(s,11) + I^2J^2G^2(s,11) + \ldots\right)JG(s,01) \\ &= IJG(s,10)\left(I - IJG(s,11)\right)^{-1}JG(s,01) \end{aligned} \tag{44}$$

where $I$ is the identity matrix. Substitution of the error transform matrices (40) and (41) into (44) and substitution of the result into (34) gives the error transform

$$t(s,I,J) = IJ^2 \frac{p_{2L}p_{2R}}{s^2 - s(p_{2L}+p_{2R}) + p_{2L}p_{2R}(1-IJ)} \frac{p_{4L}p_{4R}}{(s-p_{4L})(s-p_4R)} \tag{45}$$

Substitution of $I$=1, $J$=1 shows that some pole locationshave changed as a result of the loop in the error state graph (Fig. 3c) from $p_{2L}$and $p_{2R}$ to 0 and $p_{2L}+p_{2R}$. The series in (44) converges between these new pole values and, to distinguish this more restrictive requirement from the original ROC, we will term it the SROC (series ROC). The shift of poles and introduction of a SROC is characteristic of the transfer function bound, as discussed in more detail below.

To obtain the transform for bit error rate, we interchange differentiation and contour integration in (36). The result of differentiation and substitution of $I$=1 and $J$=1 is

$$q_b(s) = p_{2L}p_{2R} \frac{s^2 - s(p_{2L}+p_{2R}) + p_{2L}p_{2R}}{s^2(s-p_{2L}-p_{2R})^2} \frac{p_{4L}p_{4R}}{(s-p_{4L})(s-p_{4R})} \tag{46}$$

Evaluation of the residues at the RP poles and substitution of the pole values (42), (43) gives the true union bound on bit error rate as

$$P_b = \frac{P_s}{2}\left(\frac{1}{\sqrt{1+P_s}} - 1 + P_s\right) \tag{47}$$

where $P_s$ (39) is a function of the SNR and the quality of channel estimation. For useful channels, $P_s \approx 1/\Gamma_s$.

We show in Fig. 4 the true union bound (47) and simulated performance of this code for the case of perfect channel state information, i.e., $\mu$=1. Also shown are the Chernoff bound and the bound from [5]. It is apparent that the Chernoff bound is about 3.6 dB looser than the TUB. The bound in [5] is very tight for the given range of SNR, and it requires less computation than the TUB. However, it does not extend easily from perfect CSI to

differential and pilot-based detection strategies.

We show in Fig. 5 the TUB of the same code with differential detection. The normalized fade rate $f_dT$=0.05 and, following [3], the correlation between fading gains in successive intervals is $\sigma_g^2 J_0(2\pi f_d T)$. We simply make the corresponding changes (39) to $P_s$ in (47). For comparison, included on the same Figure is the bound from [4], which is seen to be approximately 2 dB looser than the TUB at an error rate of $10^{-2}$, and 1.7 times higher than the TUB in the irreducible error rate region. We also compared the two bounds at $f_dT$=0 and found that the bound from [4] is approximately 1 dB looser than the TUB.

### *B. Scalar Transfer Functions*

The error transform matrices (40)-(41) and the matrix transfer function for this code have the same values for all components, so considerable simplification can be obtained by treating them as scalars, and applying scalar labels to the error state diagram before calculating the transfer function. A sufficient condition [8] for the matrix transfer function to have equal values for all components is that, for each $\boldsymbol{e}$, the row sums (or column sums, if consistent) of its error transform matrix $\boldsymbol{G}(s,\boldsymbol{e})$ are equal. Such a code is termed uniform, and the scalar branch labels are the row sums.

A sufficient condition for uniformity in the additive white Gaussian noise channel [8] is that the signal subconstellations associated with each row are related by reflections or rotations, so that Euclidean distances are preserved. In general, this is not true for the Rayleigh fading channel, since, even when the Euclidean metric is employed, the pole locations (18) depend on constellation properties in addition to the Euclidean distance $|x-\hat{x}|$. However, for the special cases of MPSK or perfect CSI, only the Euclidean distance plays a role. Thus, under either of these conditions, any code that is uniform for the AWGN channel is also uniform for the Rayleigh fading channel. For simplicity, we have used uniform trellis codes and scalar transfer functions for all the examples in this paper.

### *C. Series Region of Convergence*

The concise representation (45) of the error transform rests on convergence of the geometric series in (44). A necessary and sufficent condition for convergence is that $|\boldsymbol{G}(s,11)|<1$; equivalently, that the elementary characteristic function $|\Phi_2(s)|<1$. Inspection of the form of such functions (17) shows that, for any $\sigma$ and $s=\sigma+j\omega$, they take their largest magnitudes at $\omega=0$. Thus, to determine convergence, it is sufficient to consider points along the real axis. For the special cases of MPSK or perfect CSI, the variable $\delta(\boldsymbol{c},\hat{\boldsymbol{c}})=0$, and the variation of the elementary characteristic function along the real axis is as shown in Fig. 6. The series convergence requirement is satisfied for $0<s<p_{2L}+p_{2R}$, which defines the SROC. Not surprisingly, the endpoints of the SROC are the new poles of the error transform (45).

The minimum point of any elementary characteristic function with $\delta(c,\hat{c})=0$ occurs midway between its poles, at $\sigma=(p_{2L}+p_{2R})/2=w$. For MPSK or perfect CSI, $w=½$, and the TUB converges at $\sigma=½$, if it converges at all. The integration line (22) is most conveniently placed if it runs through this point; equivalently, we evaluate the residues at poles to the right of $\sigma=½$.

*D. Rate 1/2, 4-State BPSK Code*

As a more complex example, we analyze the code shown in Fig. 7a, consisting of the optimal 4-state, rate 1/2 convolutional encoder [9], a parallel to serial converter, and a BPSK signal mapper. The code is uniform and the corresponding error state diagram is shown in Fig. 7b, with $\Phi_4(s)$ defined in (43).

Using the results from [9, p 245], it can be shown that the error transform $t(s,I,J)$, with $J$ set to 1, is

$$t(s,I) = \frac{\Phi_4^5(s)}{1-2\Phi_4(s)I} \tag{48}$$

The corresponding bit error probability transform is

$$q_b(s) = \frac{\partial}{\partial I} t(s,I)\Big|_{I=1} = \left(\frac{-P_s}{4}\right)^5 \frac{1}{(s-p_{4L})^3(s-p_{4R})^3(s-p_1)^2(s-p_2)^2} \tag{49}$$

where $p_{4L}$ and $p_{4R}$ are the original poles (43),

$$\begin{Bmatrix} p_1 \\ p_2 \end{Bmatrix} = \frac{1}{2}\left(1 \mp \sqrt{1-P_s}\right) \tag{50}$$

are the new poles introduced by the transfer function of the error state diagram, and the term $P_s$ was defined earlier (39). Since $P_s<1$, the two new poles introduced are real and positive. Finally the union bound $P_b$ on the bit error probability is simply $\mathbf{N}(q_b(s))$.

The SROC considerations are similar to those of the two-state code examined above. For the series giving rise to (48) to converge, we need $|\Phi_4(s)|<½$. It can be shown that this condition is satisfied when $p_1<s<p_2$, which lies within the ROC of $\Phi_4(s)$, and it consequently becomes the SROC of the error transforms $t(s,I)$ and $q_b(s)$. Because this is an MPSK code, $\sigma=½$ is in the SROC, so that the union bound $P_b$ is the sum of residues of $q_b(s)/s$ calculated at the poles $p_{4R}$ and $p_2$.

Fig. 8 shows the TUB obtained for this coded BPSK scheme operating with perfect CSI. The Chernoff bound is about 2 dB looser than our bound, but the bound in [5] is only marginally looser (0.3 dB) than the TUB. As noted earlier, [5] does not extend readily to imperfect CSI, so Fig. 9, which illustrates differential detection with $f_DT=0.05$, uses for

comparison the Chernoff bound and the bound from [4]. We see that the bound from [4] is 1dB to 2 dB looser than the TUB for BERs from $10^{-5}$ to $10^{-2}$, and that the gap increases for higher error rates.

## VI. COMPUTATION AND CONVERGENCE IN GENERAL TCM

In this Section, we address issues that arise in more general TCM schemes with multilevel constellations and imperfect CSI. We also describe the computational approach for more complex state diagrams than the ones in Section V.

### *A. Scalar Transfer Function*

Being able to use a scalar transfer function is a major simplification. We saw in Section V that, if a TCM is uniform in AWGN, it remains uniform, and consequently has a scalar transfer function, if it is an MPSK code or if we have perfect CSI. We note here that, even if the code is based on a multilevel constellation and CSI is imperfect, it remains uniform if we adopt the suboptimum Euclidean metric, as discussed in Section IIIC. Thus we must deal with a matrix transfer function only when (a) the constellation is multilevel *and* the CSI is imperfect *and* we use the optimum metric, or (b) the code is not uniform even in AWGN. All examples in this paper have scalar transfer functions.

### *B. Calculation of The Transfer Function*

Small state diagrams, such as those in Section V, can be reduced to a transfer function by inspection and elementary manipulations. The transfer function of more complex codes is obtained more easily by a formal method, using a symbolic processor. For an arbitrary uniform TCM, the error transforms $t(s,I)$ and $q_b(s)$, excluding the one-step error events in the case of codes with parallel transition, can always be written in the form

$$t(s,I) = \boldsymbol{Y}(s,I)\boldsymbol{X}(s,I)\boldsymbol{Z}(s,I) \tag{51}$$

so that

$$q_b(s) = \frac{1}{m}\left(\boldsymbol{Y}'(s)\boldsymbol{X}(s)\boldsymbol{Z}(s) + \boldsymbol{Y}(s)\boldsymbol{X}(s)\boldsymbol{H}'(s)\boldsymbol{X}(s)\boldsymbol{Z}(s) + \boldsymbol{Y}(s)\boldsymbol{X}(s)\boldsymbol{Z}'(s)\right) \tag{52}$$

where

$$\boldsymbol{X}(s,I) = \sum_{n=0}^{\infty} \boldsymbol{H}^n(s,I) = [\boldsymbol{I} - \boldsymbol{H}(s,I)]^{-1} \tag{53}$$

$\boldsymbol{H}(s,I)$ is the $(N\text{-}1)\text{x}(N\text{-}1)$ "transition" matrix of the error state diagram whose $(m,n)$th element is the row sum of the label matrix (33). The definitions for the row vector $\boldsymbol{Y}(s,I)$ and the column vector $\boldsymbol{Z}(s,I)$ are similar to $\boldsymbol{H}(s,I)$, except that the $m$th components of these matrices are associated, respectively, with the transition from the error free state $\varepsilon_0$ to the error state $\varepsilon_m$ and the transition from $\varepsilon_m$ to $\varepsilon_0$. The vectors $\boldsymbol{Y}'(s)$ and $\boldsymbol{Z}'(s)$ are obtained by taking the partial derivatives of $\boldsymbol{Y}(s,I)$ and $\boldsymbol{Z}(s,I)$ with respect to $I$ and then evaluating the results at $I$=1.

Finally, $X(s)$ and $H(s)$ are the matrices $X(s,I)$ and $H(s,I)$ evaluated at $I$=1.

Those values of $s$ for which $det[I-H(s)]=0$ become the new poles introduced by the transfer function of the error state diagram. Because of the presence of two $[I-H(s)]^{-1}$ terms in the second term on the right side of (52), these new poles are of second order. This point is well illustrated by (46) and (49).

Listed below are some tips to speed up the calculations when using a symbolic math package:

1. The quantity $P_s^{-1}$ in (39) should be rounded to the nearest integer before being substituted into the symbolic equations.
2. If possible, the squared Euclidean distance term $|d|^2$ in (38) should be expressed as a function of integers. For example, in the case of PSK, instead of using the number 0.585 to represent the squared distance between points separated by 45 degrees, use the expression $\sqrt{2}$ - 2.
3. For PSK modulations, each elementary characteristic function (17) is of the form

$$\Phi(s) = \frac{p_L p_R}{s^2 - s - p_L p_R} \tag{54}$$

Instead of using symbolic equations in $s$, replace the term $s^2$-$s$ by $\omega$ and solve for the error transform and the new poles in terms of $\omega$. Once this is done, go back and solve for $s$.

The symbolic calculation of the TUB is relatively complex and time-consuming. However, once obtained in symbolic form, evaluation for arbitrary values of SNR and quality of the channel reference is very quick, since it involves only substitution of different values of $P_s$.

### *C. Calculation of N*

Calculating the error rate involves applying the functional (22) to the error transform (52). For error transforms of low degree, it is straightforward to obtain the TUB in symbolic form. We obtain (52) symbolically and calculate the shifted poles, that is, the roots of $det[X(s)]=0$. Finally, we calculate the residues to the right of the SROC. Methods for determining the SROC are addressed below.

If the degree is uncomfortably high for symbolic manipulation, an alternative is numerical integration of (22). Through use of the Cauchy integral theorem, the infinite line integral can be converted to a finite contour integral around the appropriate poles, assuming their locations have been determined. However, we have found it preferable to perform (22) explicitly as a line integral, for two reasons. First, it is not necessary to calculate the precise pole locations, provided a point in the SROC at which to place the integration line is known. Second, the integrand decreases quickly and monotonically away from the real axis, allowing

the integration limits to be truncated to relatively small values. The resulting level of computation is significantly less than in a numerical contour integration.

*D. Convergence*

Obtaining the error rate from $q_b(s)$ (52) requires knowledge of the SROC, or a point in it, so that the correct poles or a correctly placed integration line are used in calculation of **N**. This is straightforward in the case of MPSK or perfect CSI, since $\sigma=½$ is always in the SROC. It is more difficult in the general case, so we have provided below a number of results that allow determination of the SROC or a point in it.

As a general comment, the SROC is relatively wide for normal error rates, so it is easy to determine. It shrinks as the SNR decreases and the TUB breaks away from the true error rate prior to divergence. However, when the TUB is on the edge of divergence, it is of questionable value, and simulation is more appropriate.

The first result is the observation that the SROC, if one exists, is the middle interval between poles, in the sense that it has the same number of poles to the right of it as to the left, when multiple poles are counted separately. The reason is related to the generally U-shaped behaviour of the elementary characteristic functions (the branch weights of the graph). We first note that the original ROC has an equal number of poles on either side. Next, in the examples of Section V, we saw that the shifted poles correspond to the values of $\sigma$ on either side of the SROC at which the magnitude of the branch weights in the geometric series becomes unity, causing series divergence. The SROC of those examples is therefore the middle interval. In the more general (52), the series enumerates all the cyclic paths through the graph, each of which diverges on either side of its SROC, resulting in a shifted pole on either side. Again, the SROC is the middle interval. The significance is that, after calculating the roots of $det[\boldsymbol{X}(s)]=0$, we simply evaluate residues at all poles to the right of the middle interval.

For calculation by numerical integration, we do not have to calculate the poles of (52). In this case, all we need is a single point in the SROC through which to place the integration line. Finding such a point can be as simple as scanning the original ROC (between 0 and the minimum RP pole of the elementary characteristic functions) for a real $\sigma$ for which the dominant eigenvalue of $\boldsymbol{H}(\sigma)$ is less than one in magnitude, since this is a requirement for convergence of the goemetric series.

In fact, for normal error rates, where the SROC is fairly wide, even the eigenvalue calculation can be avoided. We note that, for any real point $\sigma$ in the original ROC, the matrix $\boldsymbol{H}(\sigma)$ is a non-negative irreducible matrix with a spectral radius $r$ (magnitude of the dominant eigenvalue) that satisfies [11]

$$\sum_{\min} \le r \le \sum_{\max} \tag{55}$$

where $\Sigma_{min}$ and $\Sigma_{max}$ are, respectively, the smallest and the largest row sums of $\boldsymbol{H}(s)$. Thus, if $\Sigma_{max}<1$, the point is in the SROC, and if $\Sigma_{min}>1$, the point is not in the SROC. If neither of these conditions holds, we can calculate $\boldsymbol{H}^n(\sigma)$, where n is some small positive integer, in order to exaggerate the spectral radius, and repeat the test. We found that $n$=2 was sufficient for any interesting error rate. If no point with spectral radius less than one is found, then we can scan more finely near promising regions, or abandon the search with the conclusion that the TUB is probably close to divergence.

If, for some reason, an approximation to the entire SROC is desired, instead of just a point within it, we can generalize the above procedure as follows:

1. Set the search index $n$ to 1.
2. Calculate symbolically the matrix $\boldsymbol{H}^n(\sigma)$ and its row sums.
3. For each row sum found in step 2, solve symbolically or numerically for the range of real values $\sigma$ within which the row sum is less than one.
4. If there exists a common region of $\sigma$ within the original ROC where all row sums are less than one, then we have found a solution. If no row has a region in which its row sum is less than one, then there is no solution, and the TUB diverges. Otherwise, increase $n$ by 1 and go to step 2.

The computational effort in steps 2 and 3 increases with increasing $n$, but, as noted above, $n\leq 2$ is sufficient in the interesting region. Also note that a solution, if found, may be a subinterval of the SROC, since the algorithm is based on the upperbound on the spectral radius, while the actual value of the spectral radius can lie anywhere between the lower and upper bounds.

### *E. An Example*

We have applied these procedures to analyze Ungerboeck's 4 and 8 state 8PSK codes [10] . The error state diagram for the 4-state code, excluding the one-step error events, is shown in [8, Fig. 4.12]. For brevity, it is not reproduced here. From the error state diagram, it can be shown that the quantities $\boldsymbol{Y}(s,I)$, $\boldsymbol{H}(s,I)$ and $\boldsymbol{Z}(s,I)$ are given by

$$\boldsymbol{Y}(s,I) = \begin{bmatrix} t_1 & 0 & 0 \end{bmatrix} \tag{56}$$

$$\boldsymbol{H}(s,I) = \begin{bmatrix} 0 & t_4 & t_2 \\ t_5 & 0 & 0 \\ 0 & t_3 & t_6 \end{bmatrix} \tag{57}$$

and

$$Z(s,I) = \begin{bmatrix} 0 \\ t_7 \\ 0 \end{bmatrix} \tag{58}$$

The scalar labels $t_1$ to $t_7$ associated with the various transitions are defined in [8, p. 123], with the following substitutions:

$$\begin{aligned} W(001) &= \Phi_1(s) \\ W(010) = W(110) &= \Phi_2(s) \\ W(011) = W(111) &= \tfrac{1}{2}\left(\Phi_1(s) + \Phi_3(s)\right) \\ W(100) &= \Phi_4(s) \\ W(101) &= \Phi_3(s) \end{aligned} \tag{59}$$

where the poles of the elementary characteristic functions $\Phi_1(s)$ and $\Phi_3(s)$ are

$$\begin{Bmatrix} p_{1L} \\ p_{1R} \end{Bmatrix} = \frac{1}{2}\left(1 \mp \sqrt{1 + 4P_s/(2-\sqrt{2})}\right) \tag{60}$$

and

$$\begin{Bmatrix} p_{3L} \\ p_{3R} \end{Bmatrix} = \frac{1}{2}\left(1 \mp \sqrt{1 + 4P_s/(2+\sqrt{2})}\right) \tag{61}$$

respectively, and the poles of $\Phi_2(s)$ and $\Phi_4(s)$ were defined earlier in (42) and (43). As for Ungerboek's 8-state code, the error state diagram is given in Fig. 8 of [2], with the $\alpha_k$ defined in Table III under the following substitutions

$$\begin{aligned} W(100) &= \Phi_1(s) \\ W(010) = W(011) &= \Phi_2(s) \\ W(110) = W(111) &= \tfrac{1}{2}\left(\Phi_1(s) + \Phi_3(s)\right) \\ W(001) &= \Phi_4(s) \\ W(101) &= \Phi_3(s) \end{aligned} \tag{62}$$

The matrix $[I - H(s)]^{-1}$ and the row vector $Y(s,I)$ for this code are, respectively, the transposes of the first and second terms on the right side of (20) in [2], and the column vector $Z(s,I)$ is

$$Z(s,I) = \begin{bmatrix} 0 & \alpha_4 & 0 & \alpha_{16} & 0 & \alpha_{24} & 0 \end{bmatrix}^T \tag{63}$$

Once again, we caution the reader that the result obtained from (54)-(61) does not account for

the one-step error events caused by parallel transitions. They can be calculated in a straightforward way by applying (24) to the elementary characteristic function (17), with distances obtained from the uncoded constellation resulting from set partitioning.

We evaluated the TUBs for these codes using the methods described earlier in this Section. Because they are MPSK codes, the point $\sigma=\frac{1}{2}$ is in the SROC. For the 4-state codes, we symbolically found the poles and evaluated the residues to the right of $\sigma=\frac{1}{2}$. For the 8-state code, it was more convenient to apply (22) to (52) numerically, with Gauss integration along a vertical line through $\sigma=\frac{1}{2}$. The matrix inversion of (52) was performed numerically at each step. Convergence of the integration was fast and monotonic.

The resulting TUBs for perfect CSI, including coded and uncoded bits, are shown in Figs. 10 and 11, respectively. Once again, the results show that the bound from [5] is only slightly looser than the TUB. However, as mentioned earlier, it is not readily applicable to differential detection or other forms of imperfect CSI, in contrast to the method of this paper. Accordingly, Fig. 12, which illustrates the 8-state code with differential detection, uses the bound from [4] and a simulation for comparison. We see that the bound from [4] is considerably looser than the TUB, especially at low SNR and in the error floor region. Comparison with the simulation results, on the other hand, shows that the TUB is quite accurate in the static channel, except for low SNR values, but is well above the true values in fast fading, especially in the error floor region.

## VII. CONCLUSIONS

We have presented in this paper a method of combining the exact pairwise error probability with the transfer function in order to obtain union bounds for trellis-coded modulations. The method is based on the observation that the branch labels in the error state diagram of a TCM scheme can be replaced by the corresponding elementary characteristic functions of the branch error events. This means the transfer function of the error state diagram is now the "characteristic function" of the union bound on the error probability, and the sum of residues of this function, calculated at proper pole locations, yields the exact union bound itself.

Our bound is very tight, as confirmed by simulations and comparisons with other bounds in the literature. In fact, it is unlikely that a tighter bound can be obtained, unless a method can be found to decorrelate some of the error events in the union bound. It is also very general, since it can be applied to coherent, partially coherent and pilot-based detection, to diversity reception, and to QAM and PSK types of constellations. One drawback of our method, though, is that is requires more computation than those reported in the literature, such as [4] and [5]. However, with the speed of present-day computers and availability of powerful symbolic software packages, the computational complexity is no longer a serious concern.

## REFERENCES


[1] D. Divsalar and M.K. Simon, "The Design of trellis coded MPSK for fading channels: Performance criteria", *IEEE Trans. Commun.*, vol. COM-36, pp. 1004-1012, Sept. 1988.

[2] R. G. McKay, P.J. McLane, and E. Biglieri, "Error bounds for trellis coded MPSK on a fading mobile satellite channel", *IEEE Trans. Commun.*, vol. 39, pp. 1750-1761, Dec 1991.

[3] J.K. Cavers and P. Ho, "Analysis of the error performance of trellis coded modulations in Rayleigh fading channels", *IEEE Trans. Commun.*, vol. 40, pp. 74-80, Jan 1992.

[4] K.Y. Chan and A. Bateman, "The performance of reference based M-ary PSK with trellis coded modulation in Rayleigh fading", *IEEE Trans. Vehicul. Techn.*, vol. 41, pp. 190-198, May 1992.

[5] S.B. Slimane and T. Le-Ngoc, "Tight bounds on the error probability of coded modulation schemes in Rayleigh fading channels", *IEEE Trans. Vehicul. Techn.*, vol. 44, pp. 121-130, Feb. 1995.

[6] J.G. Proakis, *Digital Communications, 3rd Edition*, New York, McGraw Hill, 1995.

[7] M. Schwartz, W. Bennett, and S. Stein, *Communication Systems and Techniques*, New York, McGraw Hill, 1966.

[8] E. Biglieri, D. Divsalar, P.J. McLane, and M.K. Simon, *Introduction to Trellis-coded Modulation with Applications*, New York, Macmillan Publishing Company, 1991.

[9] A. Viterbi and J.K. Omura, *Principles of Digital Communication and Coding*, New York, McGraw Hill, 1979.

[10] G. Ungerboeck, "Channel coding with multilevel/phase signals" , *IEEE Trans. Inf. Th.*, vol. IT-28, pp. 55-67, Jan 1982.

[11] F. R. Gantmacher, *Matrix Theory, Vol. 2*, Chelsea Publishing Company, New York, N.Y., 1959.

LIST OF ILLUSTRATIONS

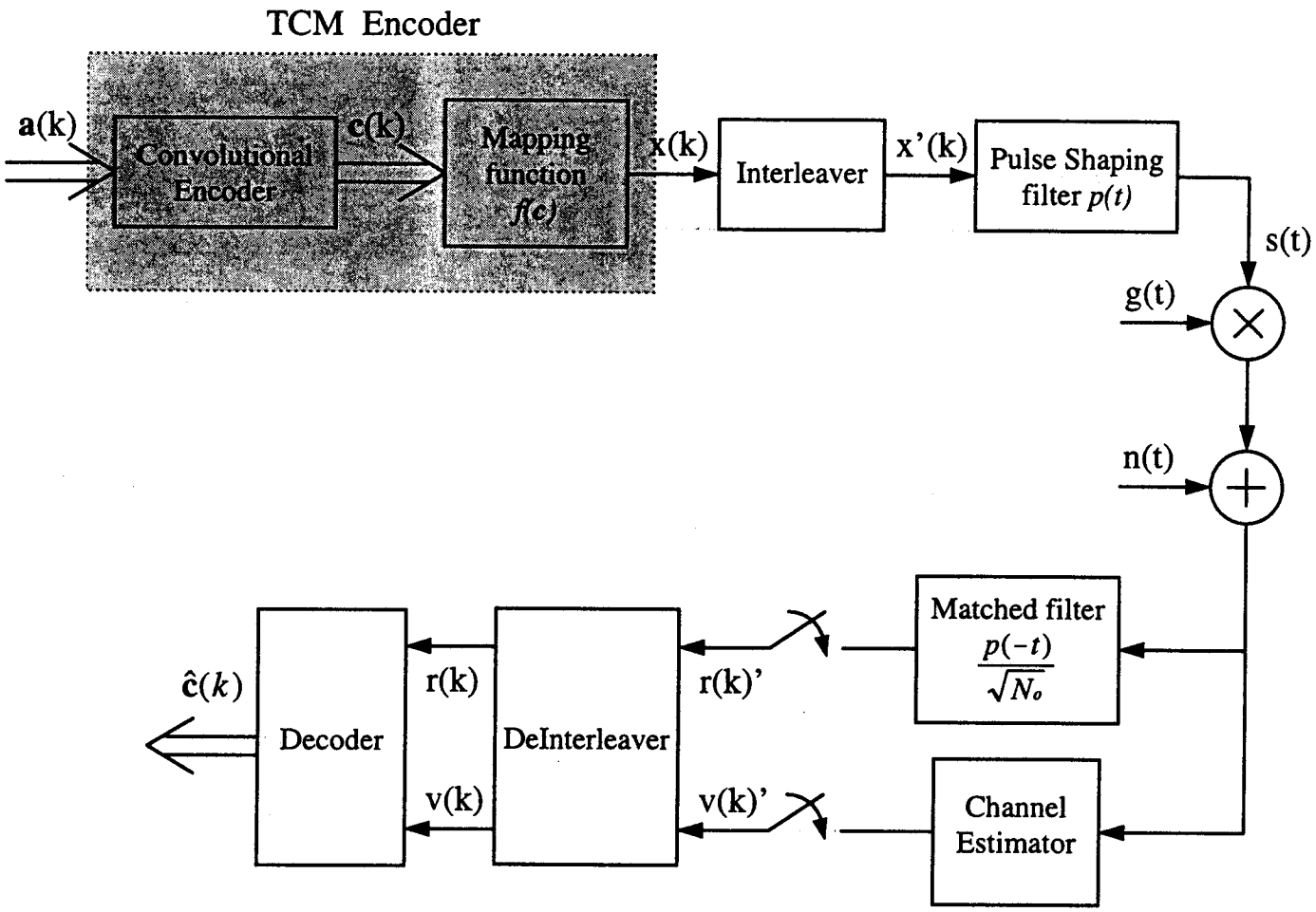



Figure 1: System Block Diagram.

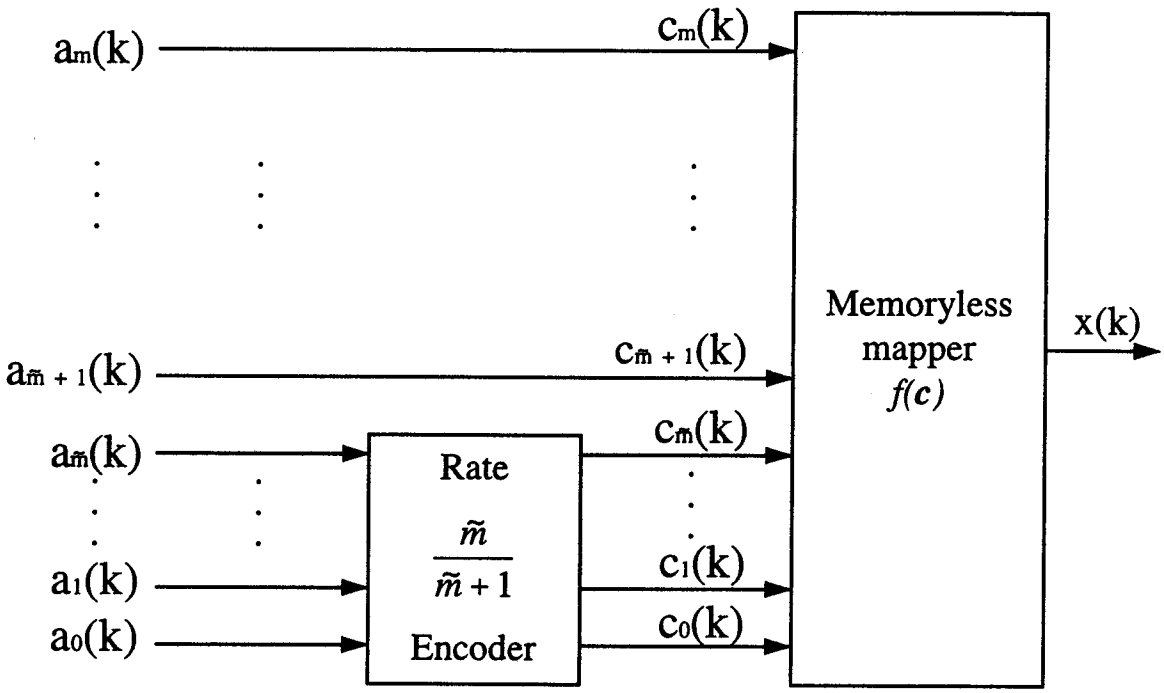
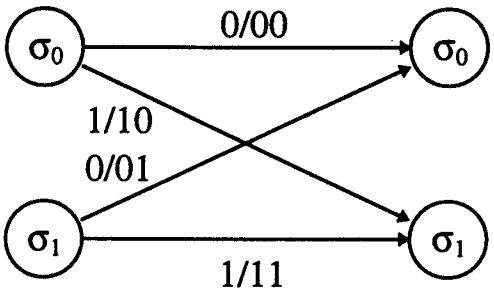
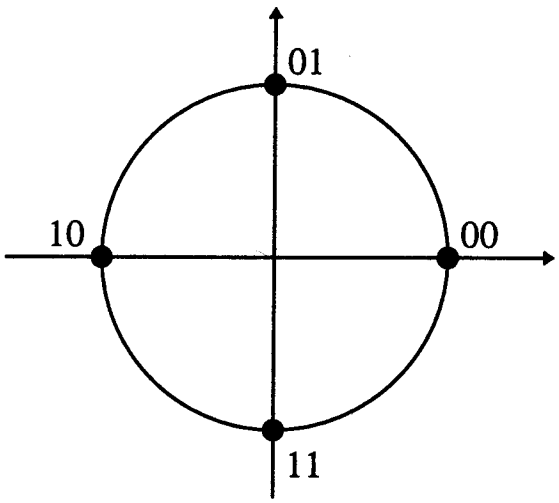


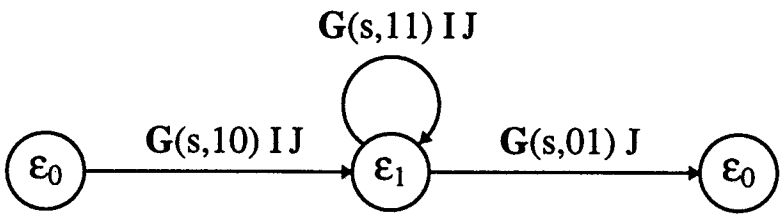

Figure 2: General Structure for TCM Schemes.




(a)




(b)




(c)

Figure 3: The 2-state QPSK Code (a) Trellis, (b) Signal Constellation, (c) Error State Diagram.

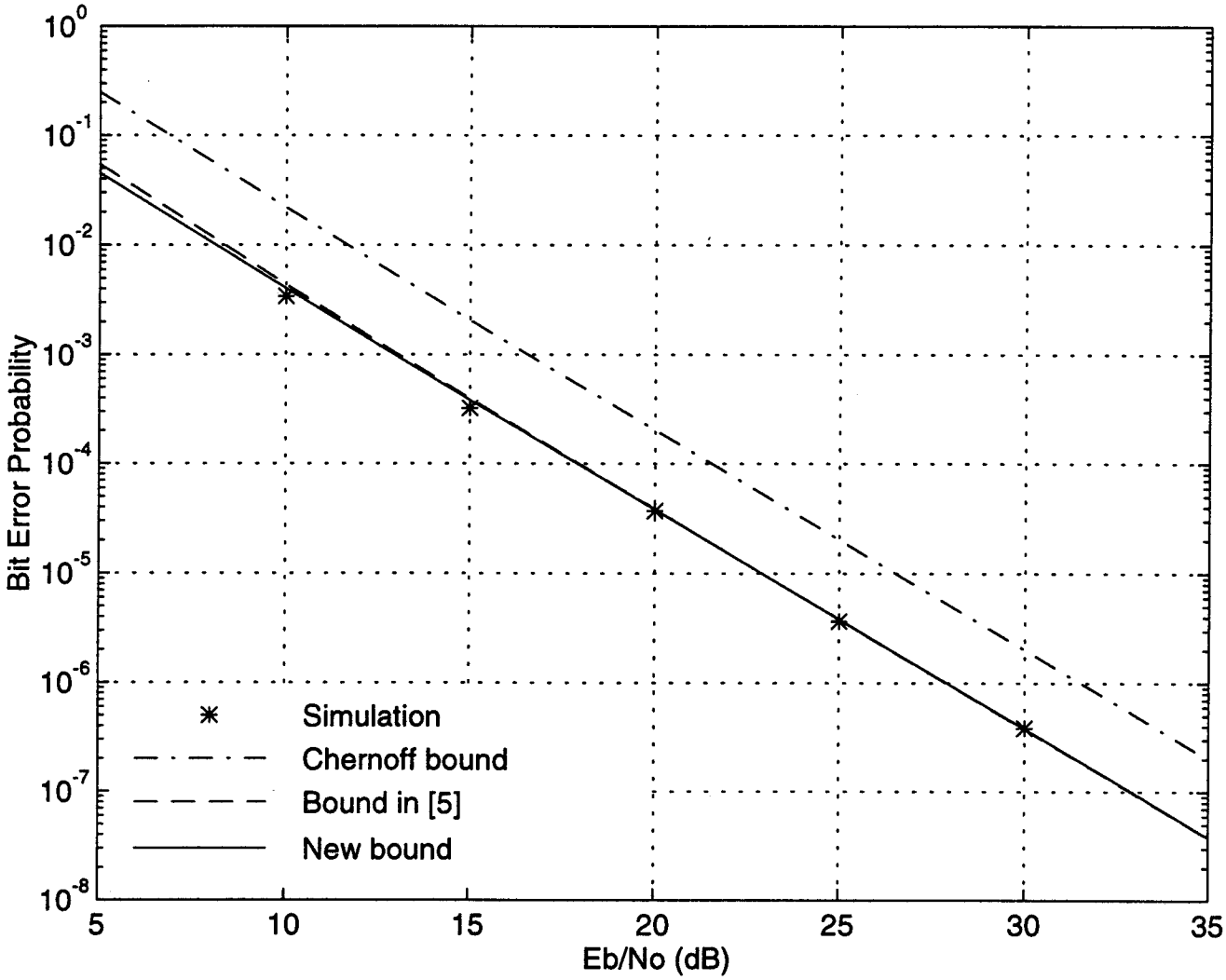



Fig. 4 Upper Bounds and Simulation Results for the 2-state QPSK Code in Rayleigh Fading with Perfect CSI.

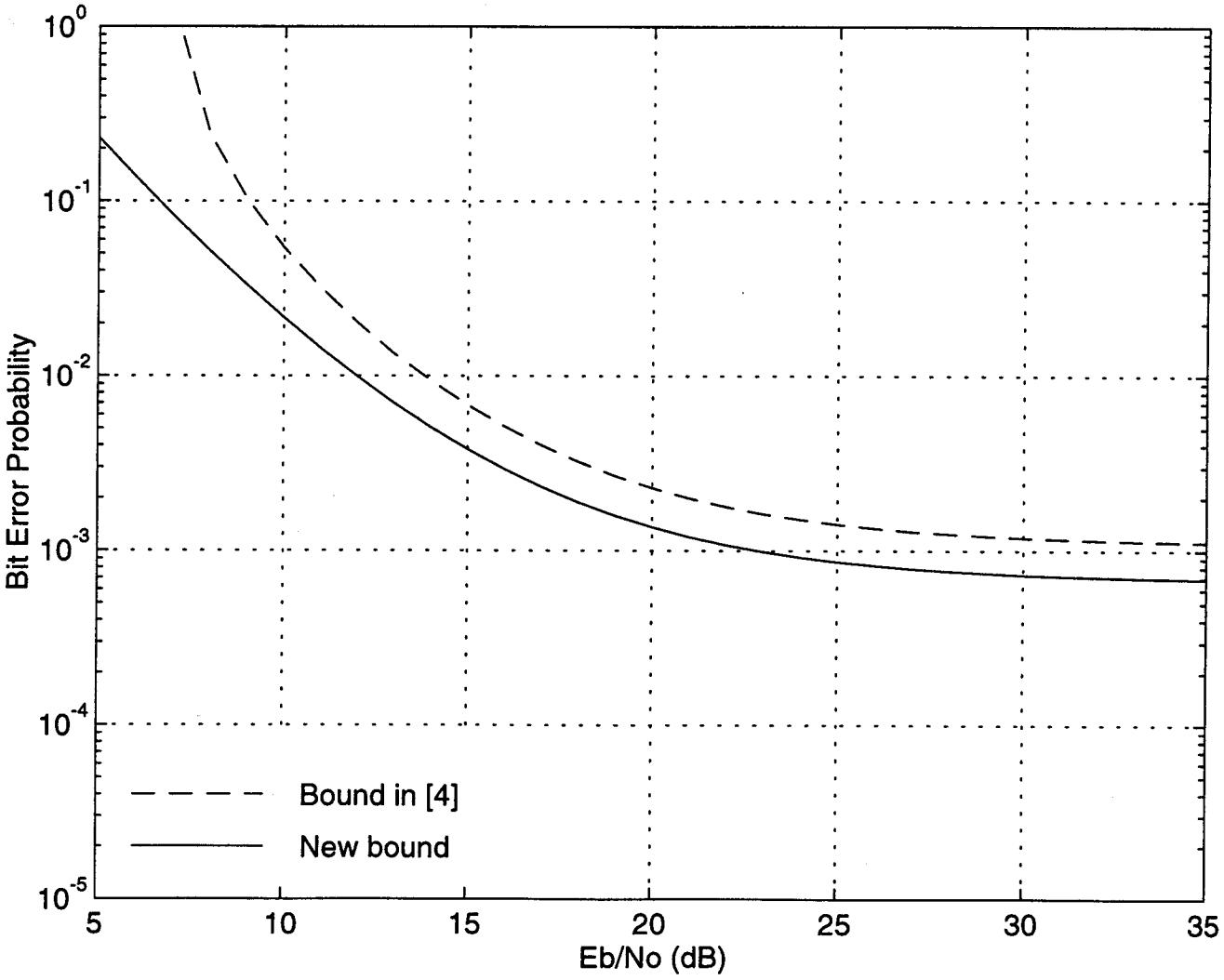



Fig. 5 Upper Bounds for the 2-state QPSK Code in Rayleigh Fading with Differential Detection.

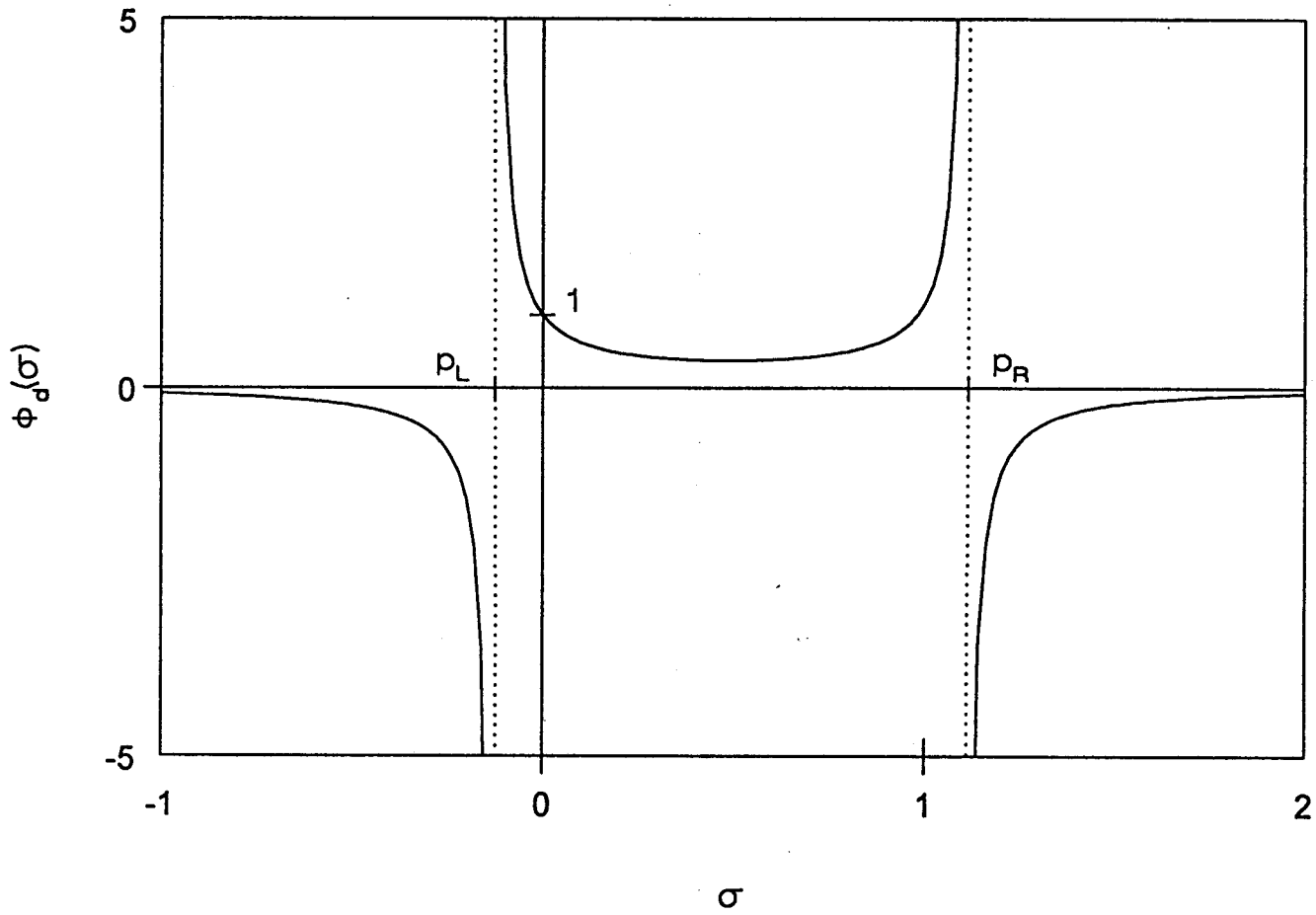
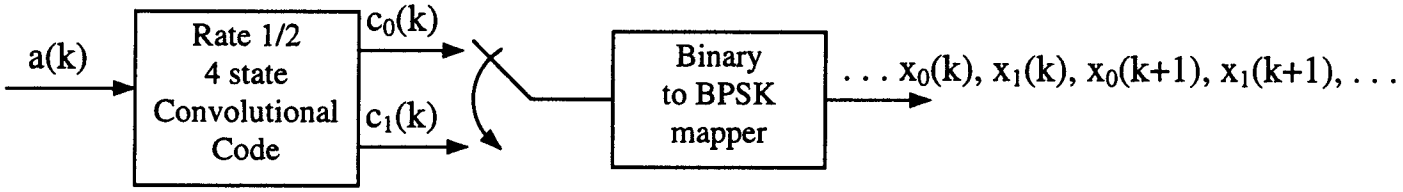
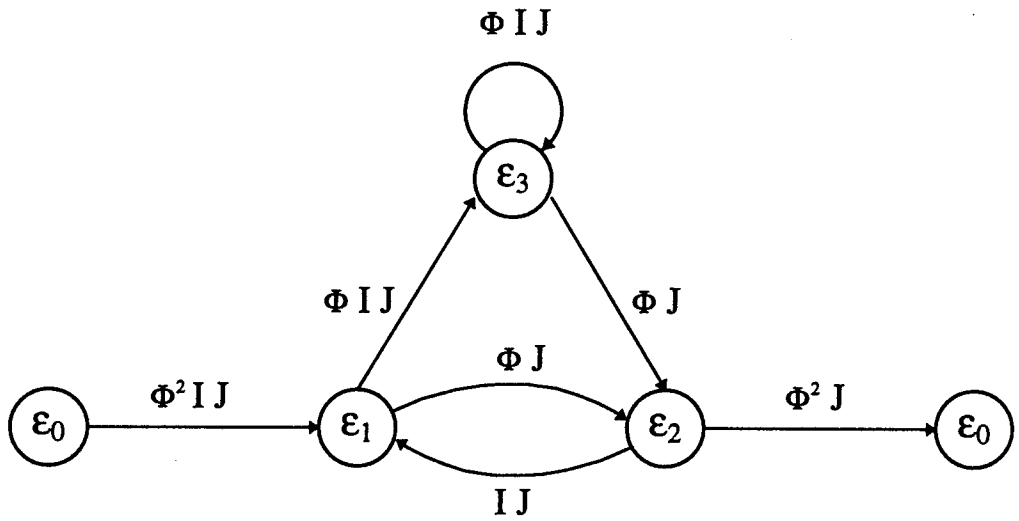



Fig. 6 Behaviour of Elementary Characteristic Function on the Real Axis.




(a)




(b)

Figure 7: The Rate 1/2, 4-state BPSK Code : (a) Block Diagram, (b) Error State Diagram.

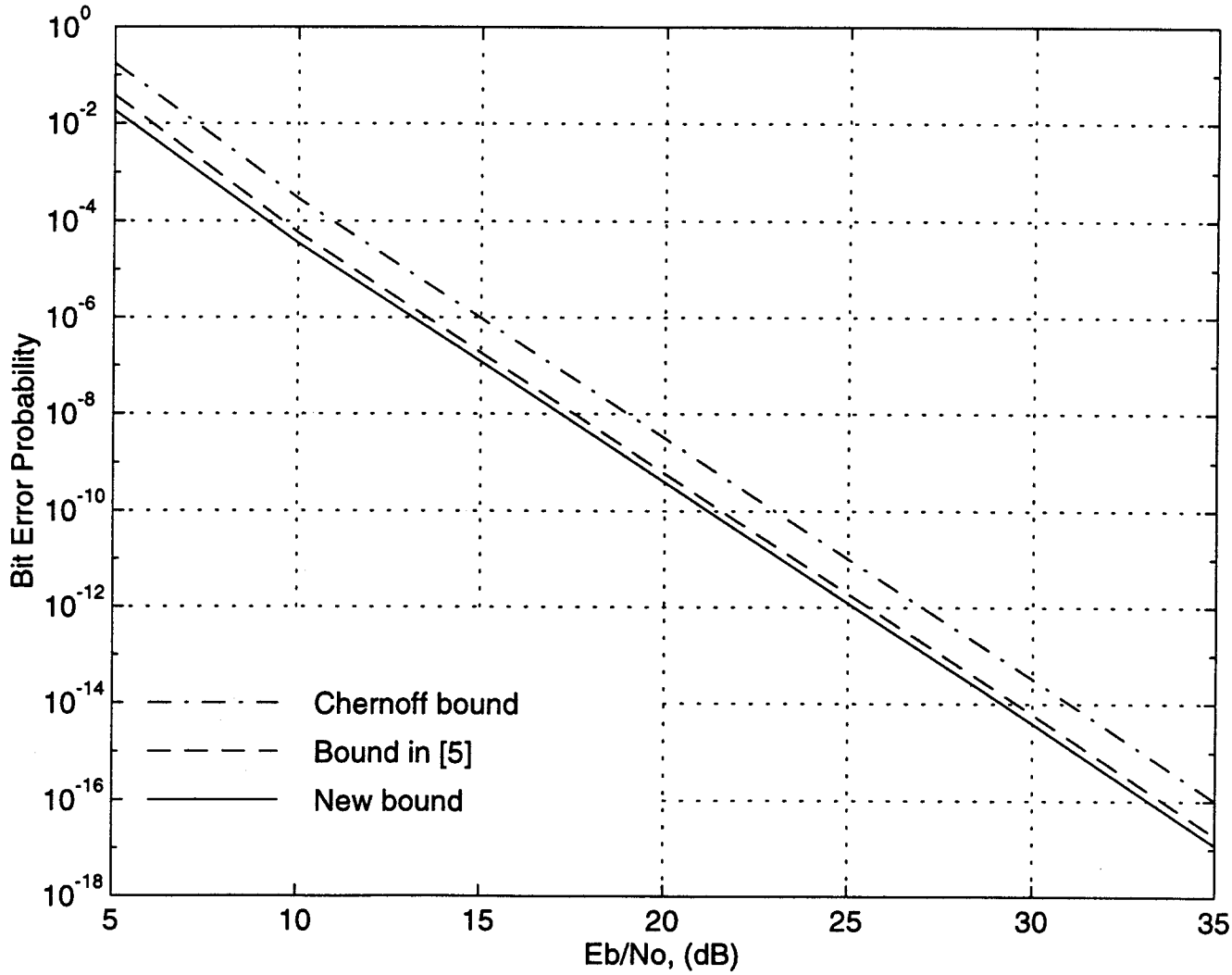



Fig. 8 Upper Bounds for the 4-state BPSK Code in Rayleigh Fading with Perfect CSI.

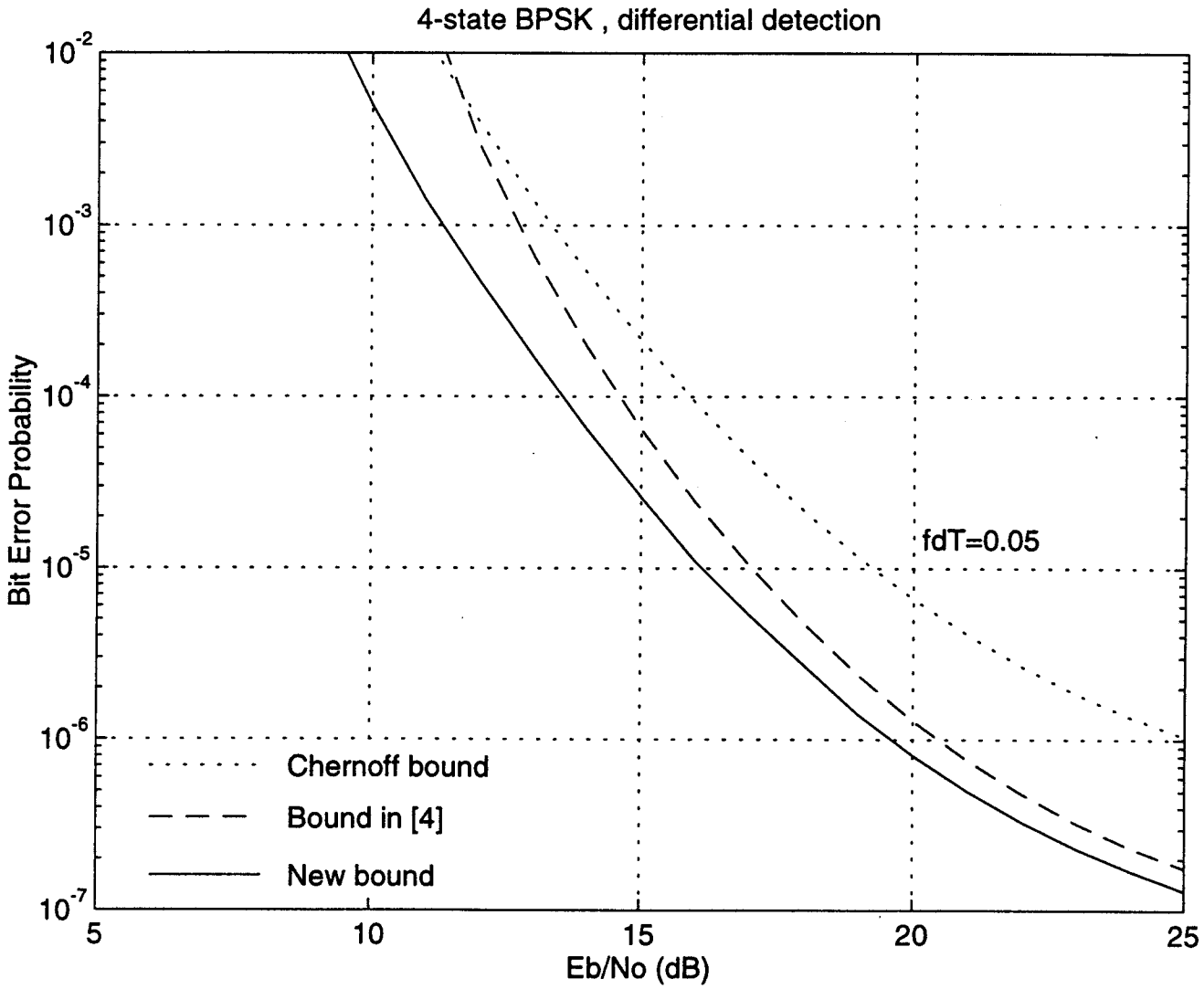



Fig. 9 Upper Bounds for the 4-state BPSK Code in Rayleigh Fading with Differential Detection.

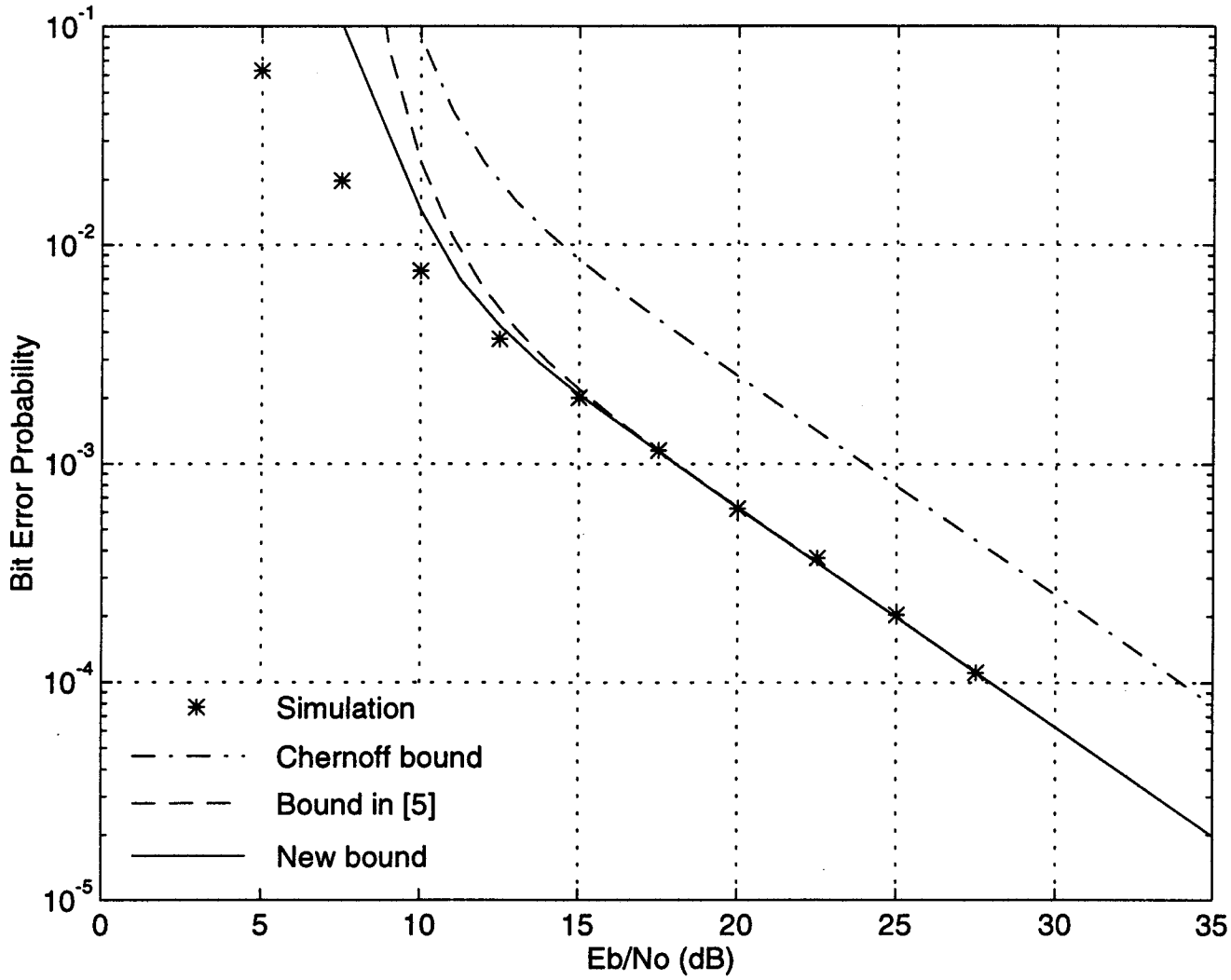



Fig. 10 Upper Bounds and Simulation Results for Ungerboeck's 4-state Code in Rayleigh Fading with Perfect CSI.

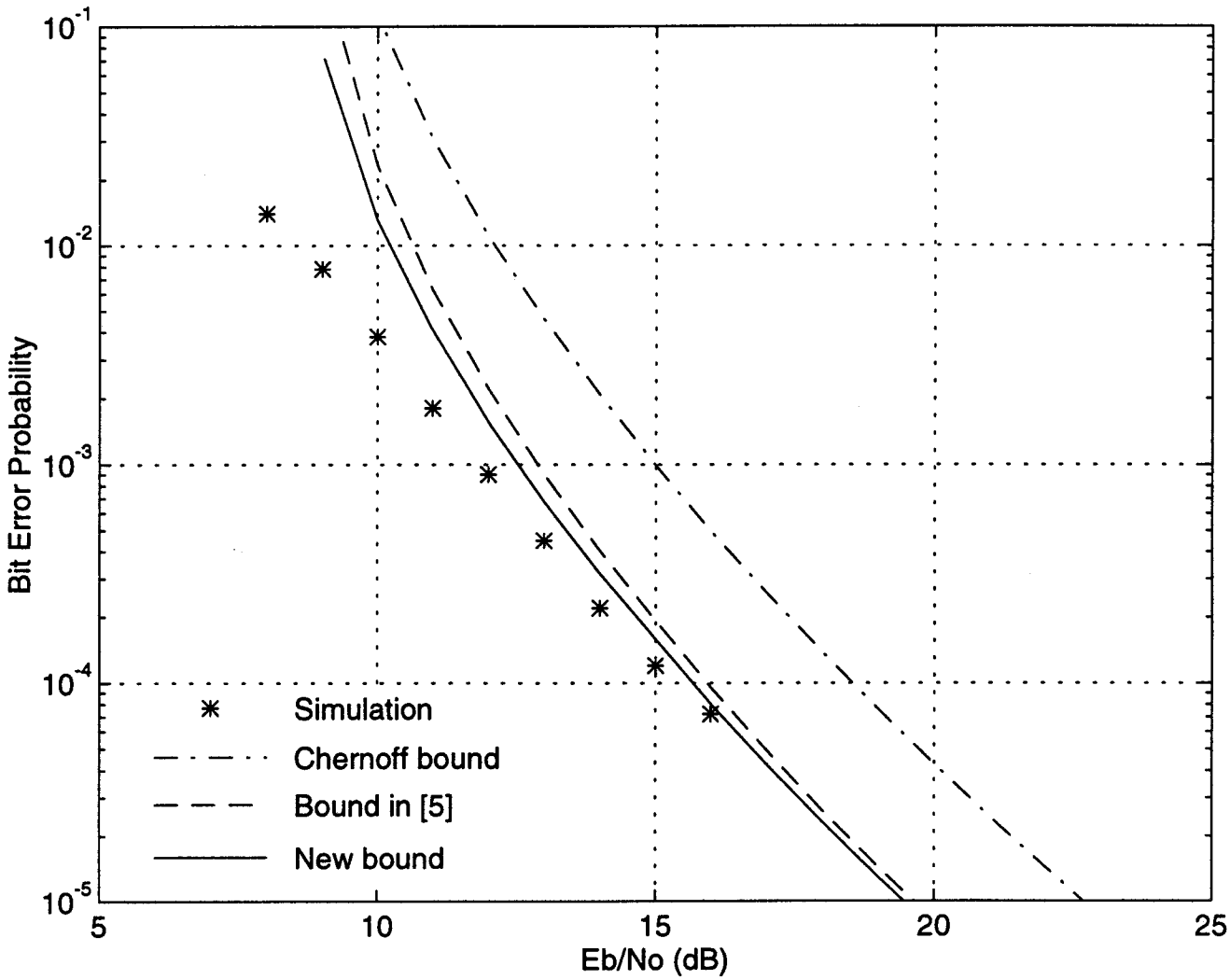



Fig. 11 Upper Bounds and Simulation Results for Ungerboeck's 8-state Code in Rayleigh Fading with Perfect CSI.

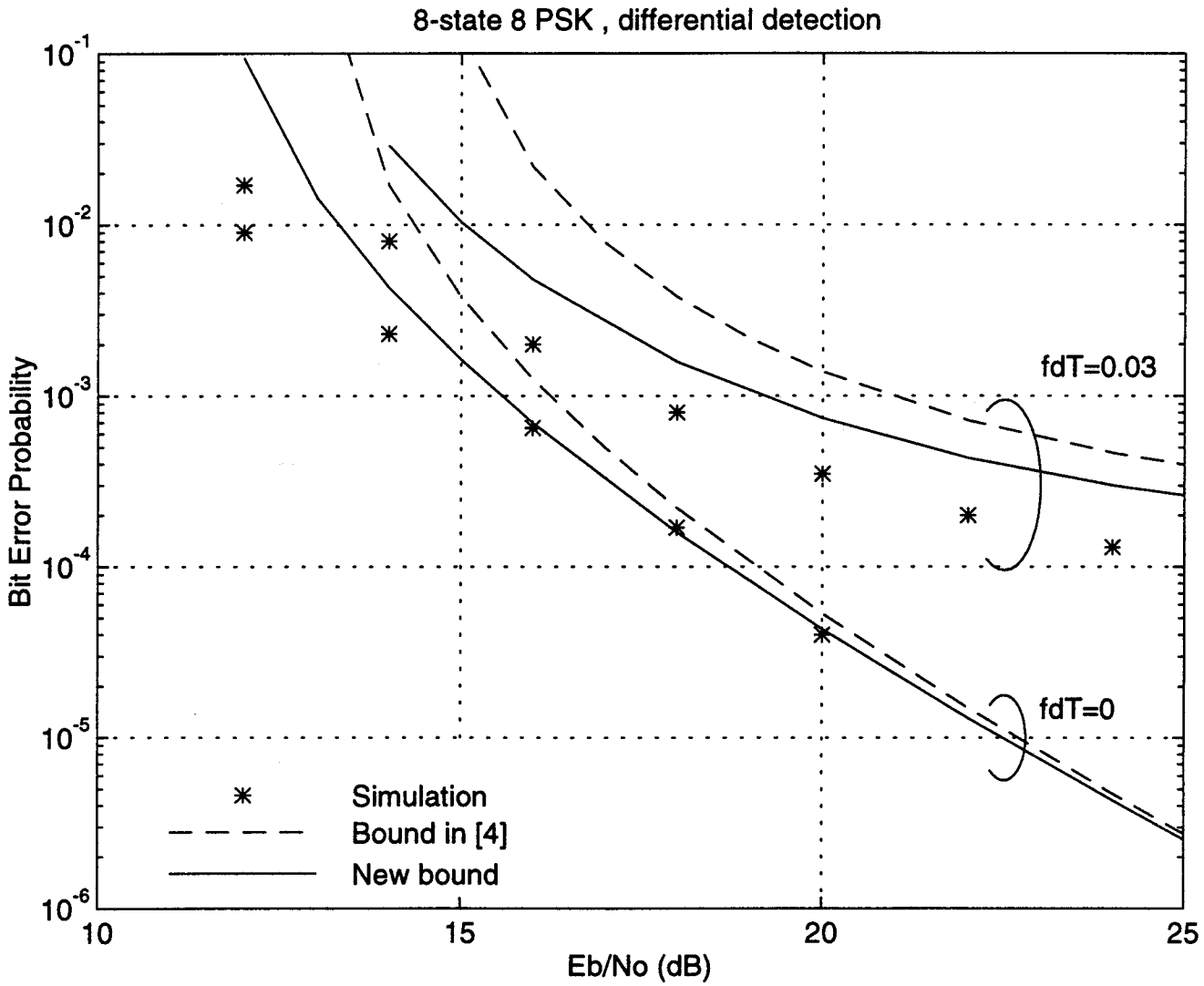



Fig. 12 Upper Bounds and Simulation Results for Ungerboeck's 8-state Code in Rayleigh Fading with Differential Detection.