A Hybrid-ARQ System Using Rate-Compatible Trellis Codes Designed for Rayleigh Fading

by M. Eröz and T. Fuja

T.R. 96-75

ISR
INSTITUTE FOR SYSTEMS RESEARCH

*Sponsored by
the National Science Foundation
Engineering Research Center Program,
the University of Maryland,
Harvard University,
and Industry*

A Hybrid-ARQ System Using Rate-Compatible Trellis Codes Designed for Rayleigh Fading

Mustafa Eröz and Tom Fuja [*]
Electrical Engineering Department
Institute for Systems Research
University of Maryland
College Park, MD 20742

October 8, 1996

Abstract

This paper presents classes of rate-compatible trellis codes designed for channels with flat, slow Rayleigh fading. The codes thus described are "multiple" TCM (MTCM) codes as proposed by Divsalar and Simon – i.e., codes in which multiple symbols are associated with each transition through the trellis; by applying appropriate puncturing tables to low-rate MTCM codes, we obtain families of MTCM codes, all of which can be decoded with (essentially) the same decoder.

By means of computer search, several such families are designed so that each family member is at least as good as any comparable code in the literature. ("Good" here is defined in terms of minimum time diversity and minimum squared product distance, the most important parameters for performance over Rayleigh fading channels.)

A protocol to implement these rate-compatible trellis codes in a type-II hybrid ARQ format with only a low-rate feedback channel is described. Upper bounds on the resulting bit error rate are developed and the results are used to select the best adaptive code from several possibilities. Simulation results comparing the proposed scheme with fixed-rate codes of the same throughput show substantial coding gain. Finally, a protocol modification limiting the variability of the code rate over a frame is described; this modification eliminates the need for excessive buffering, with a very small effect on performance.

Key Words: Trellis-coded modulation, fading channels, Rayleigh channels, automatic repeat request, type-II hybrid ARQ.

[*]This work was supported by NSF grant NCR-8957623 and by the NSF Engineering Research Centers Program, CDR-8803012. It was presented in part at MILCOM '95, San Diego, CA, November 5-8, 1995.

1 Introduction and Background

Trellis coded modulation (TCM) is a combined strategy for channel coding and modulation that can provide significant gain for bandlimited channels. Although TCM was originally introduced for additive white Gaussian noise (AWGN) channels, there has been considerable interest recently in applying TCM to the fading channels that characterize wireless communication systems.

Consider a lowpass-equivalent channel in which the complex-valued output sequence $\{y_i\}$ is related to the complex-valued input $\{x_i\}$ as follows:

$$y_i = \rho_i x_i + n_i. \tag{1}$$

Here n_i is a zero-mean complex Gaussian vector, with independent real and complex parts, each with variance $N_0/2$. The random variable ρ_i is the *fading* associated with the i^{th} transmitted symbol; for channels with no line-of-sight between transmitter and receiver – e.g., land mobile radio channels – $\{\rho_i\}$ is well modeled by a sequence of Rayleigh random variables. With sufficient interleaving, successive fading values may be assumed to be independent.

It has been shown [1] that, at high SNR, the most important design parameters for trellis codes over a channel with independent Rayleigh fading are the *minimum time diversity* L and the *minimum squared product distance* d_p^2. The minimum time diversity of a code is the minimum Hamming distance (in channel symbols) between two distinct encoded sequences; if $\mathcal{C}$ is the set of channel input sequences corresponding to valid trellis paths, and if $d(\cdot,\cdot)$ denotes Hamming distance, then

$$L = \min\{d(\mathbf{c},\mathbf{c}') : \mathbf{c},\mathbf{c}' \in \mathcal{C}, \mathbf{c} \neq \mathbf{c}'\}.$$

The corresponding minimum squared product distance is given by

$$d_p^2 = \min\{ \prod_{\{i:c_i \neq c_i'\}} |c_i - c_i'|^2 : d(\mathbf{c},\mathbf{c}') = L\}.$$

Note that this is unlike AWGN channels, where the most important design parameter is the minimum squared Euclidean distance between valid paths.

Divsalar and Simon [2] proposed *multiple trellis-coded modulation* (MTCM) for fading channels; MTCM codes have $k \geq 2$ transmitted symbols associated with each trellis branch, thus permitting a time diversity of k between parallel branches.

Although trellis codes with large L and d_p^2 can yield coding gains over fading channels, they do not exploit the channel's time varying nature; adapting the code to changing channel conditions may provide even more gain. Various researchers [6]–[8] have used this idea of adaptive code selection – in which "incremental redundancy" is requested via a feedback channel when the channel is sufficiently noisy – to increase robustness. Some have used punctured codes [9] in this framework; puncturing a code means deleting some redundant symbols, thereby increasing the rate and decreasing its robustness. Such codes can be used in an automatic repeat request (ARQ) format by using a high-rate (punctured) code when the channel is good and using the lower-rate (unpunctured) code when the channel is bad. Hagenauer [8] proposed using a whole family of "rate-compatible" punctured convolutional codes in this way. A family of codes is said to be rate-compatible if each code in the family is a punctured version of all the lower-rate codes in the family; rate-compatible punctured codes provides the system designer with a family of codes with different rate/performance profiles – each of which can be decoded using (essentially) the same Viterbi decoder.

This paper extends the idea of rate-compatibility to MTCM codes. This is done by finding families of rate-compatible MTCM codes that are optimal for Rayleigh fading channels — i.e., with L and d_p^2 maximized. We then propose a protocol for these codes that allows the transmission of incremental redundancy while achieving a specified average rate.

In Section 2, we review the Divsalar/Simon design procedure for MTCM codes; we then use this procedure to design families of rate-compatible MTCM codes. In Section 3, we use the codes in an ARQ format and derive upper bounds on their performance. We then show how to use these bounds to select parameters of the adaptive code. Finally, we modify the protocol to limit the variability of the rate, thereby lessening the buffer requirements.

2 Rate Compatible TCM for Fading Channels

Our goal is to design a family of rate-compatible trellis codes for the Rayleigh fading channel using the multiple TCM design approach of [2]. We begin by reviewing that approach.

2.1 Multiple TCM for Fading Channels

The modulation is M-ary phase-shift keying (M-PSK), with signal set $\mathcal{S} = \{s_0, s_1, \ldots, s_{M-1}\}$, where $s_i = \cos(i2\pi/M) + j\sin(i2\pi/M)$.

First consider codes with fully-connected trellises and M parallel branches; M different branches (with M different branch labels) connect every pair of states. Let k denote a code's multiplicity – i.e., each branch label consists of k channel symbols, denoting the k transmitted symbols associated with that transition. The rate (or *throughput*) of such a code is determined by the number of states; if there are 2^ν trellis states, then there are $M \cdot 2^\nu$ outgoing edges in each state, meaning a rate of $[\nu + \log_2(M)]/k$ bits per symbol.

Associated with each "bundle" of M parallel branches is a set of M branch labels – M different k-tuples over $\mathcal{S}$. Let A_i denote the i^{th} set of branch labels; then, generalizing slightly the notation in [2], we express A_i as follows:

$$A_i = \{(s_j, s_{n_1 j \oplus c_{i,1}}, s_{n_2 j \oplus c_{i,2}} \ldots s_{n_{k-1} j \oplus c_{i,k-1}}) : j = 0, 1, \ldots M - 1\}, \qquad (2)$$

where the subscript operations are performed modulo M. We assign such a set of branch labels to each bundle of parallel branches between every pair of states; thus, for each state we need 2^ν sets – one for each outgoing bundle. We will require two such collections of sets – one associated with the even-numbered states and the other associated with the odd-numbered states. (See Figure 1.) As a result, we require $2^{\nu+1}$ sets of labels $\{A_i : i = 0, 1, \ldots, 2^{\nu+1} - 1\}$.

The choice of the multiplicative constants $\{n_\ell\}$ and the additive constants $\{c_{i,\ell}\}$ determine the distance properties of the resulting codes:

- The same $k-1$ multiplicative constants $\{n_\ell : 1 \le \ell \le k-1\}$ are used in all the A_i's. Thus the choice of the n_ℓ's affects the distance properties *within* each A_i but does not affect those *between* two different A_i's – i.e., the n_ℓ's affect the distances associated with parallel branches.

- The additive constants $\{c_{i,\ell} : 0 \leq i \leq 2^{\nu+1} - 1, 1 \leq \ell \leq k-1\}$ do not vary (for a given coordinate) within each A_i; thus they have no effect on the distance properties *within* each set but instead control the distances *between* them – i.e., the $c_{i,\ell}$'s affect the distance properties associated with non-parallel paths through the trellis.

Note that, because the multiplicative constants control *only* distance properties associated with parallel paths and the additive constants control *only* distance properties associated with non-parallel paths, we may search for the optimal values independently.

Divsalar and Simon showed that the multiplicative constants in Table 1(a) for the case $k = 2$ are optimal. (More than one entry in the table means that any such entry is optimal.) The optimal multiplicative constants for $k = 3$ and $k = 4$ are given in Tables 1(b) and 1(c).

2.2 Rate-Compatible TCM for Fading Channels

Examination of Table 1 reveals that, for each M, the optimal multiplicative constant $n_1^\star$ for $k = 2$ also appears as one of the optimal constants $(n_1^\star n_2^\star)$ for $k = 3$ – and that these are, in fact, two of the optimal constants $(n_1^\star n_2^\star n_3^\star)$ for $k = 4$. So, at least for these values of k and M, we can find a family of rate-compatible MTCM codes where each code has optimal multiplicative constants. The procedure for choosing the additive constants will be made clear later in the section, but let us first give an example of a rate-compatible MTCM code.

Example 1: Figure 1 shows the trellis for a $k = 3$ code using 8-PSK modulation. (In this trellis diagram, the signal s_i is represented by i.) For each transition there are $M = 8$ parallel branches; each set of eight 3-tuples next to a state contains the labels for the corresponding "bundle" of parallel branches. (For example: The eight branches from state "0" to state "1" are labeled with $\{s_0s_2s_7, s_1s_5s_0, \ldots, s_7s_7s_6\}$). This code is obtained by constructing the sets A_i for $i = 1, 2, \ldots, 8$ according to equation (2) with the optimal multiplicative coefficients $(n_1n_2) = (31)$ and appropriate additive constants; for example, $A_0 =$

$\{s_0s_0s_0, s_1s_3s_1, s_2s_6s_2, s_3s_1s_3, s_4s_4s_4, s_5s_7s_5, s_6s_2s_6, s_7s_5s_7\}$ is constructed using $c_{0,1} = c_{0,2} = 0$ while $A_1 = \{s_0s_2s_7, s_1s_5s_0, s_2s_0s_1, s_3s_3s_2, s_4s_6s_3, s_5s_1s_4, s_6s_4s_5, s_7s_7s_6\}$ is constructed using $c_{1,1} = 2$ and $c_{1,2} = 7$. If we now puncture the last symbol on each branch, the result is an optimal $k = 2$ code with multiplicative coefficient $n_1 = 3$. □

Table 1(d) contains the optimal multiplicative constants for $k = 5$. For $M \geq 16$, none of the optimal constants from Table 1(c) appear as a subset of the corresponding entry in Table 1(d). So if the signal constellation has cardinality $M \geq 16$ then one cannot find a pair of optimal rate-compatible MTCM codes (of this design) of multiplicities four and five. For $M \leq 8$, however – for BPSK, QPSK, and octal PSK modulation – our search results indicate that one can always find optimal multiplicative constants at every multiplicity $k \leq 10$.

Now consider the additive constants – $\{\mathbf{c}_{i,\ell} : \ell = 1, \ldots, k-1\}$ for each set A_i. Recall that the additive constants are the same in each coordinate within a set A_i, and they thus control the distance properties between different A_i's; recall also that the additive constants may be optimized without regard to the multiplicative constants.

We choose the additive constants to maximize the following distances (in decreasing order of importance) between any pair of signals diverging from or remerging into the same state:

- minimum diversity;
- minimum squared product distance;
- minimum squared Euclidean distance.

In all of the codes that we have searched, we were able to find additive constants such that the diversity between any pair of k-tuples that diverge from or remerge into a common state is at least $k - 1$, the best that can be achieved. After optimizing the minimum diversity between such branches, we then maximized the minimum squared product distance; for codes with multiplicity $k \geq 3$, this maximization does not affect the minimum squared product distance of the code since the minimum diversity (and hence the minimum squared product distance) of the code is determined by parallel branches. Our motivation for this and the squared Euclidean distance optimization was mainly to enhance performance at low-to-moderate SNR.

This search for optimum additive constants may be computationally intensive. For an MTCM code of multiplicity k that makes use of v different sets $(A_i, i = 0, 1, \ldots, v-1)$, we must search for $v(k-1)$ different additive constants, each of which can take any value between 0 and $M-1$. Moreover for each combination of these constants, we are required to examine all possible pairs of k-tuples that diverge from or remerge into a common state. However, the search effort can be simplified by noting that, for a given k-tuple $\mathbf{c} \in A_i$ generated by equation (2), the set of distances between $\mathbf{c}$ and all the elements of $A_{i'}$ $(i' \neq i)$ does not depend on which element of A_i is selected. (For details, see [12].)

The results of our search through five families of rate-compatible trellis codes are given in Table 2. The trellis structures we assume are shown in Figure 2; in each case there are 2^ν states in the fully-connected trellis and $2^{\nu+1}$ sets of branch labels, indicated by $\{B_0, B_1, \ldots, B_{2^{\nu+1}-1}\}$. The B_i's are generated as follows:

- For codes in which there are M parallel branches, where M is the alphabet size – i.e., codes for which the throughput is $\nu + \log_2(M)$ bits per symbol – each of the $2^{\nu+1}$ B_i's contain M labels and are the same as the A_i's as defined in equation (2).

- For codes in which there are $M/2$ parallel branches – i.e., codes for which the throughput is $\nu + \log_2(M) - 1$ bits per symbol – each of the B_i's contain $M/2$ labels and are obtained by generating 2^ν different A_i's according to equation (2) and then "splitting" each A_i into two. The even entries of A_i become B_{2i} and the odd entries of A_i become B_{2i+1}, for $i = 0, 1, \ldots, 2^\nu - 1$.

Table 2 contains the multiplicative and additive constants necessary to construct the A_i's (and thus the B_i's). The first digit in each triple of constants corresponds to the $k = 2$ code; the first two digits correspond to the $k = 3$ code, and all three digits correspond to the $k = 4$ code. For the multiplicative constants, only one such triple is necessary; while for the additive constants we require one triple for each A_i – $2^{\nu+1}$ triples when there are M parallel branches and 2^{ν} triples when there are $M/2$ parallel branches.

Although each code in Table 2 is rate-compatible with the other codes in the same family, each code has distance properties *at least* as good as any code in the literature with the same number of states, throughput, and multiplicity; the rate-compatibility property "costs" nothing in terms of performance for these codes over Rayleigh fading channels. Some comparisons are made in Table 3.

3 A Hybrid-ARQ System

When the transmission characteristics of a communication channel change over time, a fixed-rate error control code might not provide consistently acceptable performance. Under such circumstances, adapting the power of the code to changing channel conditions may be necessary; the rate compatible MTCM codes described in Section 2 are well-suited to this use. In this section we describe a protocol in which these codes are used in a type-II hybrid ARQ format, with incremental redundancy used as needed when the fading becomes excessive.

3.1 Description of the Protocol

The communication system that we consider is shown in Figure 3. The demodulator and Viterbi decoder are assumed to have perfect channel state information (CSI) – i.e., the fading values $\{\rho_i\}$ in equation (1); techniques such as pilot symbol insertion [4] or decision feedback coupled with adaptive linear prediction [5] can be employed to recover the CSI. We also assume that the demodulator achieves coherent detection.

As shown in the figure, the feedback comes from the demodulator rather than an error detection device after the Viterbi decoder, as it would in a "typical" ARQ scheme. The decision to ask for additional redundancy is based on the perceived channel state rather than any detected errors. When the received symbols are sufficiently faded, a decoder error is anticipated; so instead of wasting time trying to decode and (most likely) detecting an error, the receiver asks for more redundancy as soon as the fade is perceived.

Example 2: The mother code is described by the trellis in Figure 1. Since each state has $2^\nu = 32$ outgoing branches and each branch is associated with $k = 3$ channel symbols, the bandwidth efficiency of this code is $\log_2(32)/3 = 1.67$ bits/symbol.

When the fading conditions for the first two symbols of a branch are unfavorable, the encoder sends the third symbol; when the fading conditions are favorable, the encoder does not send the third symbol – i.e., the third symbol is "punctured" – thereby in effect creating a code with rate $5/2 = 2.5$ bits/symbol. The receiver declares that the fading conditions are unfavorable if the fade value falls below a threshold for at least one of the first two symbols; otherwise the channel conditions are considered favorable.

As an example, suppose the encoder passes through the state sequence

$$0 \to 1 \to 2 \to 3 \to 1 \to 0 \to 0.$$

Suppose further that the signal 3-tuples associated with the encoded sequence is given by

$$s_0s_2s_7 \quad s_6s_1s_4 \quad s_1s_7s_3 \quad s_0s_3s_4 \quad s_1s_6s_5 \quad s_0s_0s_0$$

The transmitter – depending on the feedback from the receiver – sometimes sends all three symbols on a branch and sometimes only the first two symbols. We now explain how to implement this using a low capacity feedback channel.

In a typical communication system for fading channels, symbols are interleaved prior to transmission and de-interleaved prior to decoding at the receiver; this is done to "break up" the fades and make them appear independent at the decoder. To illustrate, assume a 3×6 interleaver array, placing the above symbols in the array as shown.

$$\begin{bmatrix} s_0 & s_6 & s_2 & s_1 & s_7 & s_4 \\ s_1 & s_0 & s_7 & s_3 & s_3 & s_4 \\ s_1 & s_0 & s_6 & s_0 & s_5 & s_0 \end{bmatrix}$$

Our strategy for reading the encoded sequence into the interleaver array is as follows:

- Each row of the array contains two branch labels.

- The first two symbols in each label – the ones that are *always* transmitted – are placed in the rows "round robin" style. For instance, the first row of the array contains the first symbol from the first label (s_0), followed by the first symbol from the second label (s_6), followed by the second symbol from the first label (s_2), followed by the second symbol from the second label (s_1).

- Once all the "required" symbols are placed in the row, we next place the "optional" symbols – i.e., the third symbols from the labels, the ones that may be punctured. For instance, the first row of the array finishes with the third symbol from the first label (s_7) and the third symbol from the second label (s_4).

The symbols are transmitted column-by-column. All the symbols in the first four columns are always sent, while some of the symbols from the last two columns may be punctured. For instance, if the receiver observes that one or both of the first two symbols from the first branch were severely faded – i.e., s_0 and s_2 – then the receiver tells the transmitter that the last symbol (s_7 in this case) *should* be transmitted; otherwise the last symbol is punctured. Note that this feedback channel requires relatively low capacity – for this example, six bits of feedback for each frame of data in the forward channel. Moreover, substantial delay is acceptable in this protocol; between the time the receiver knows if it needs the third symbol from a label and the time the transmitter must send (or *not* send) that symbol, there are approximately N signaling intervals, where N is the interleaver depth (i.e., the number of rows in the array). If necessary, the delay can be increased by delaying the optional symbols one or more frames – e.g., in this example, the third symbol on each branch is sent (or not sent) F frames after the first two symbols from that branch.

The average throughput R_{avg} is uniquely determined by the fading threshold T_d that triggers the request for incremental redundancy. By varying T_d, we can achieve any throughput between that of the mother code and that of the punctured code. Assuming that two consecutive symbols on a branch are subject to independent fading, and letting $p(T_d) = P(\rho > T_d)$ be the probability that a particular fading value exceeds T_d, then

$$p^2(T_d)\frac{1}{R_p} + (1 - p^2(T_d))\frac{1}{R_m} = \frac{1}{R_{\text{avg}}}.$$

If ρ is Rayleigh-distributed normalized so that $E[\rho^2] = 1$, then $p(T_d) = e^{-T_d^2}$ and so

$$T_d = \sqrt{(1/2)\ln\frac{R_{\text{avg}}(R_p - R_m)}{R_p(R_{\text{avg}} - R_m)}}$$

If we choose $R_m = 1.67$, $R_p = 2.5$, and $R_{\text{avg}} = 2$ bits/symbol then $T_d = \sqrt{\ln(2)/2} \approx 0.589$.

To compare the coding gain of our adaptive code with that of a fixed-rate TCM scheme optimized for fading channels, we take an eight–state, 8–PSK code with a throughput of 2 bits/symbol and distance parameters, $L = 2$, $d_p^2 = 8$ [1]. This code requires the same number of comparisons to decode one information bit as the adaptive code [12].

In Figure 4, we give simulation results for both adaptive and fixed–rate codes over a Rayleigh fading channel with normalized bandwith $B_D T_s = 0.01$; this corresponds to a vehicle speed of 60 MPH, a carrier frequency of 900 MHz, and a symbol rate of 8000 symbols/sec. A 50 × 15 block interleaver is used. The results show that the adaptive code provides a coding gain of more than 3 dB over the fixed-rate code at a BER of 10^{-5}.

Example 3: The trellis of the mother code is given in Figure 5. This code has a multiplicity of $k = 4$, a throughput of $R_4 = 0.75$ bits/symbol, and a time diversity of $L_4 = 4$; if the last symbol on each branch is punctured, the result is a multiplicity-three code with rate $R_3 = 1.0$ bits/symbol and diversity $L_3 = 3$. If the last *two* symbols on each branch are punctured, the result is a multiplicity-two code with a throughput of $R_2 = 1.5$ bits/symbol and a diversity of $L_2 = 2$. All these codes are two-state codes with QPSK modulation, and, taken together, form the first family of codes in Table 2. Our protocol will use all three codes.

The transmitter sends the first two symbols per branch regardless of the channel conditions. If there is no deep fade on either of these two transmitted symbols – i.e., both of the fade values are above a threshold T_1 – then the transmitter sends no more symbols from that branch; if either of the first two symbols are faded below T_1, then the third symbol is sent. A similar process occurs after the third symbol is received: if all three received fade amplitudes are above a threshold T_2, the transmitter punctures the fourth symbol; otherwise the fourth (and last) symbol is sent. (Obviously, $T_1 > T_2$.) The values of T_1 and T_2 determine the average rate of the system. However, now the average rate does *not* uniquely determine T_1 and T_2. Any values of T_1 and T_2 satisfying the "average throughput equation" achieves the desired throughput. To determine which operating point minimizes the bit-error rate, we develop, in the next section, upper bounds on the bit error rate of these rate-compatible MTCM schemes.

3.2 Performance of Rate-Compatible MTCM Codes in a Hybrid-ARQ Environment

In this section, we develop upper bounds in the bit error rate of a rate-compatible trellis structure consisting of three nested codes of multiplicities two, three, and four. The generalization to larger families of codes is straightforward.

We first briefly review Chernoff bounding techniques applied to the bit error rate of fixed-rate TCM over fading channels. In Section 3.2.2, we extend this analysis to our rate-compatible MTCM structure. In Section 3.2.3, we discuss how to choose the thresholds to minimize BER while maintaining a specified average throughput. Finally simulation results are provided to check the tightness of the bounds and to assess the resulting coding gain.

3.2.1 An Upper Bound to the Bit Error Rate of TCM Over Fading Channels

To establish notation, we briefly review the Chernoff bound on the BER of a coherent fixed-rate TCM system in Rayleigh fading. The approach is the one in Biglieri *et al* [3].

Recall that the received complex baseband signal y_k at time k is given by $y_k = \rho_k x_k + n_k$ where x_k and n_k are the complex baseband representations of the transmitted signal and the noise (respectively), and ρ_k is the magnitude of the fade. Since coherent detection is assumed, ρ_k is real-valued; moreover, we assume the sequence $\{\rho_k\}$ can be perfectly estimated at the receiver. For a land mobile radio channel, ρ_k is Rayleigh distributed, with pdf

$$f_{\rho_k}(\rho) = \begin{cases} 2\rho\exp(-\rho^2); & \text{if } \rho > 0 \\ 0 & \text{if } \rho \leq 0. \end{cases}$$

Over an observation window of L signal intervals, a maximum likelihood decoder chooses as its estimate of the transmitted sequence that sequence $\mathbf{X}_L = (x_1, x_2, ..., x_L)$ minimizing $\sum_{i=1}^{L} |y_i - \rho_i x_i|^2$. Then the conditional pairwise error probability is given by

$$P(\mathbf{X_L} \rightarrow \hat{\mathbf{X}}_\mathbf{L} \,|\, \underline{\rho}_\mathbf{L}) = P(\sum_{i=1}^{L} |y_i - \rho_i x_i|^2 > \sum_{i=1}^{L} |y_i - \rho_i \hat{x}_i|^2 | \underline{\rho}_\mathbf{L})$$

where $\hat{\mathbf{X}}_L = (\hat{x}_1, \hat{x}_2, ...\hat{x}_L)$ and $\underline{\rho}_\mathbf{L} = (\rho_1, \rho_2 ... \rho_L)$. Using the Chernoff bound, this conditional pairwise error probability can be upper bounded as [3]

$$P(\mathbf{X_L} \rightarrow \hat{\mathbf{X}}_\mathbf{L} | \underline{\rho}_\mathbf{L}) \leq \prod_{i=1}^{L} \exp(-\frac{1}{4N_o} \rho_i^2 |x_i - \hat{x}_i|^2)$$

where $N_0/2$ is the variance of the real and imaginary components of n_k.

For fixed-rate TCM, assuming independent consecutive fades, the unconditioned pairwise error probability can be computed as follows:

$$\begin{aligned} P(\mathbf{X_L} \rightarrow \hat{\mathbf{X}}_\mathbf{L}) &\leq \prod_{i=1}^{L} E_{\rho_i} \left[\exp(-\frac{1}{4N_o} \rho_i^2 |x_i - \hat{x}_i|^2) \right] \\ &= \prod_{i=1}^{L} \frac{1}{1 + \frac{1}{4N_o} |x_i - \hat{x}_i|^2}. \end{aligned} \tag{3}$$

This can be used to upper bound the error event probability via the union bound:

$$\begin{aligned} P(e) &\leq \sum_{L=1}^{\infty} \sum_{\mathbf{X}_L} \sum_{\hat{\mathbf{X}}_L \neq \mathbf{X}_L} P(\mathbf{X}_L) P(\mathbf{X}_L \rightarrow \hat{\mathbf{X}}_L) \\ &\leq \sum_{L=1}^{\infty} \sum_{\mathbf{X}_L} \sum_{\hat{\mathbf{X}}_L \neq \mathbf{X}_L} 2^{-KL-\nu} \prod_{i=1}^{L} \frac{1}{1 + \frac{1}{4N_o} |x_i - \hat{x}_i|^2} \end{aligned}$$

where 2^ν is the number of trellis states and K is the number of information bits per branch.

This upper bound on the error event probability can be expressed as

$$P(e) \leq \bar{T} = \frac{1}{2^\nu} \mathbf{1}^T \bar{G} \mathbf{1},$$

where, $\mathbf{1}$ is the $1 \times 2^\nu$ all-1 vector and $\bar{G}$ is the (averaged) transfer function matrix, a $2^\nu \times 2^\nu$ matrix given by

$$\bar{G} = \sum_{L=1}^{\infty} \sum_{\mathbf{E_L} \neq 0} \bar{G}(\mathbf{E}_L),$$

where $\mathbf{E}_L = (\mathbf{e}_1, \mathbf{e}_2, ... \mathbf{e}_L)$ represents an error label sequence,

$$\bar{G}(\mathbf{E}_L) = \bar{H}(\mathbf{e}_1)\,\bar{H}(\mathbf{e}_2)\,...\bar{H}(\mathbf{e}_L),$$

and the $(p,q)^{th}$ entry of $\bar{H}(\mathbf{e}_i)$ is given by

$$[\bar{H}(\mathbf{e}_i)]_{p,q} = \frac{1}{2^K} \sum_{\mathbf{c}:p\to q} \frac{1}{1 + \frac{1}{4N_0}|f(\mathbf{c}) - f(\mathbf{c}+\mathbf{e})|^2}.$$

Here, $\mathbf{c}$ takes on all values of the binary labels associated with a transition from state p to state q; the summation accounts for parallel transitions. The function $f(\cdot)$ maps the binary labels to the channel symbols.

This error event probability bound can be used to bound the bit error probability [3].

3.2.2 Upper Bound to Bit Error Rate of a Rate-Compatible TCM Structure over Fading Channels

Recall the protocol described in Example 3: For each trellis branch, either two, three, or four symbols are transmitted; let $\{\rho_i : 1 \leq i \leq 4\}$ denote the fading values imposed by the channel during the (at most) four transmissions. Then the protocol uses the values of the ρ_i's to determine how many symbols are sent. Specifically:

- If $\rho_1 > T_1$ and $\rho_2 > T_1$, then no more symbols are transmitted.
- If $\rho_1 < T_1$ or $\rho_2 < T_1$ (or both), then the third symbol is transmitted.
- Once the third symbol is transmitted, the values of ρ_1, ρ_2 and ρ_3 are compared with a second threshold T_2 ($T_2 < T_1$); if $\min(\rho_1, \rho_2, \rho_3) > T_2$, then no more symbols are transmitted – i.e., the multiplicity-three code is used. If $\min(\rho_1, \rho_2, \rho_3) < T_2$, then the fourth branch symbol is transmitted – i.e., the multiplicity-four code is used.

One can characterize the code used during a branch in terms of several disjoint events involving the fading values:

- The multiplicity-two code is used if and only $(\rho_1 > T_1)$ and $(\rho_2 > T_1)$;

- The multiplicity-three code is used if and only if one of three disjoint events occurs:
  - $(\rho_1 > T_1)$ and $(T_2 < \rho_2 < T_1)$ and $(\rho_3 > T_2)$;
  - $(T_2 < \rho_1 < T_1)$ and $(\rho_2 > T_1)$ and $(\rho_3 > T_2)$;
  - $(T_2 < \rho_1 < T_1)$ and $(T_2 < \rho_2 < T_1)$ and $(\rho_3 > T_2)$.
- The multiplicity-four code is used if and only if one of four disjoint events occurs:
  - $(\rho_1 < T_2)$;
  - $(\rho_1 > T_2)$ and $(\rho_2 < T_2)$;
  - $(T_2 < \rho_1 < T_1)$ and $(\rho_2 > T_2)$ and $(\rho_3 < T_2)$;
  - $(\rho_1 > T_1)$ and $(T_2 < \rho_2 < T_1)$ and $(\rho_3 < T_2)$.

It will be useful to refer to these events by the intervals associated with the corresponding fading values; to this end, define the following sets:

$$\begin{aligned}
\Theta_{2,1} &= \{(T_1,\infty),(T_1,\infty)\} \\
\Theta_{3,1} &= \{(T_1,\infty),(T_2,T_1),(T_2,\infty)\} \\
\Theta_{3,2} &= \{(T_2,T_1),(T_1,\infty),(T_2,\infty)\} \\
\Theta_{3,3} &= \{(T_2,T_1),(T_2,T_1),(T_2,\infty)\} \\
\Theta_{4,1} &= \{(0,T_2),(0,\infty),(0,\infty),(0,\infty)\} \\
\Theta_{4,2} &= \{(T_2,\infty),(0,T_2),(0,\infty),(0,\infty)\} \\
\Theta_{4,3} &= \{(T_2,T_1),(T_2,\infty),(0,T_2),(0,\infty)\} \\
\Theta_{4,4} &= \{(T_1,\infty),(T_2,T_1),(0,T_2),(0,\infty)\}.
\end{aligned}$$

Our analysis will be similar to that for the fixed-rate case, but the time-varying trellis complicates things. Consider a single stage of the trellis; during that stage the transmitter will send two, three, or four symbols to convey a fixed number of information bits. Moreover, we assume an error free feedback channel so that both transmitter and receiver agree on how many symbols are transmitted on each branch.

When the receiver makes an error in decoding a particular branch, there is an associated error in the binary label; let E denote such an error – i.e., the signals associated with (say) binary label C were transmitted, but the decoder chooses as its estimate the signals associated with binary label $C + E$. We compute $\bar{H}(E)$, a $2^{\nu} \times 2^{\nu}$ matrix whose $(p,q)^{\text{th}}$ element is an upper bound on the probability that a transition from state p to state q is incorrectly decoded, yielding a label error E. Because the number of symbols transmitted per branch varies, the number of bits needed to describe E varies; for QPSK, for instance, E is represented by four, six, or eight bits.

Let $E^{(2)} = (\mathbf{e}_1, \mathbf{e}_2)$, $E^{(3)} = (\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3)$, and $E^{(4)} = (\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3, \mathbf{e}_4)$ denote the specific error labels of multiplicity two, three, and four corresponding to the "generic" error label E. Then

$$\bar{H}(E) = \bar{H}_2(E^{(2)}) + \bar{H}_3(E^{(3)}) + \bar{H}_4(E^{(4)}),$$

where each element of $\bar{H}_j(E^{(j)})$ is an upper bound obtained by averaging over those fading values that result in the use of the multiplicity-j code – i.e., the fading values lying in the intervals described by the Θ's above; for instance

$$[\bar{H}_2(E^{(2)})]_{p,q} = \frac{1}{2^K} \sum_{\substack{(\mathbf{c}_1,\mathbf{c}_2): \\ p \to q}} Z_{T_1,\infty}(f(\mathbf{c}_1), f(\mathbf{c}_1 + \mathbf{e}_1)) \cdot Z_{T_1,\infty}(f(\mathbf{c}_2), f(\mathbf{c}_2 + \mathbf{e}_2)).$$

Here,

$$Z_{A,B}(x, \hat{x}) = \int_A^B (2\rho \exp(-\rho^2)) \exp(-\frac{\rho^2}{4N_0}|x - \hat{x}|^2) d\rho,$$

and $(\mathbf{c}_1, \mathbf{c}_2)$ takes on all possible label values associated with a transition from state p to state q in the multiplicity-two code. In a similar fashion,

$$[\bar{H}_3(E^{(3)})]_{p,q} = \frac{1}{2^K} \sum_{\substack{(\mathbf{c}_1,\mathbf{c}_2,\mathbf{c}_3): \\ p \to q}} \sum_{\ell=1}^{3} \prod_{j=1}^{3} Z_{A_{\ell,j},B_{\ell,j}}(f(\mathbf{c}_j), f(\mathbf{c}_j + \mathbf{e}_j)),$$

where $(A_{\ell,j}, B_{\ell,j})$ is the j^{th} element of $\Theta_{3,\ell}$, and

$$[\bar{H}_4(E^{(4)})]_{p,q} = \frac{1}{2^K} \sum_{\substack{(\mathbf{c}_1,\mathbf{c}_2,\mathbf{c}_3,\mathbf{c}_4): \\ p \to q}} \sum_{\ell=1}^{4} \prod_{j=1}^{4} Z_{A_{\ell,j},B_{\ell,j}}(f(\mathbf{c}_j), f(\mathbf{c}_j + \mathbf{e}_j)),$$

where $(A_{\ell,j}, B_{\ell,j})$ is now the j^{th} element of $\Theta_{4,\ell}$.

Now let $\mathbf{E}_{\tilde{L}} = (E_1, E_2, \ldots, E_{\tilde{L}})$ denote a sequence of error labels corresponding to an error event lasting $\tilde{L}$ branches (not symbols). Then

$$\bar{G}(\mathbf{E}_{\tilde{L}}) = \bar{H}(E_1)\bar{H}(E_2)\ldots\bar{H}(E_{\tilde{L}})$$

is a $2^{\nu} \times 2^{\nu}$ matrix whose $(p,q)^{th}$ element is an upper bound on the probability that the error event $\mathbf{E}_{\tilde{L}}$ begins in state p at any time t and ends in state q, $\tilde{L}$ branches later. From this, we can proceed as before, yielding an upper bound on the error event probability:

$$P(e) \leq \frac{1}{2^{\nu}}\mathbf{1}^T\bar{G}\mathbf{1} \qquad \text{where} \qquad \bar{G} = \sum_{\tilde{L}=1}^{\infty} \sum_{\mathbf{E}_{\tilde{L}} \neq 0} \bar{G}(\mathbf{E}_{\tilde{L}}).$$

Example 4: Consider the simple mother code of multiplicity four in Figure 6. Our adaptive scheme is composed of three codes of multiplicity $k = 2, 3,$ and 4. It is easily seen that only three binary label errors are possible during each transition:

- When two paths diverge, a label error $\xi_1 = [01, 01, 01, 01]$ occurs.

- When two paths remerge, a label error $\xi_2 = [10, 10, 10, 10]$ occurs.

- For every other branch in an error event, a label error $\xi_3 = [11, 11, 11, 11]$ occurs.

The averaged transfer function is given by

$$\bar{G} = \bar{H}(\xi_1)[I_{2\times 2} - \bar{H}(\xi_2)]^{-1}\bar{H}(\xi_3).$$

We now give an explicit expression for $\bar{H}(\xi_1)$. ($\bar{H}(\xi_2)$ and $\bar{H}(\xi_3)$ are derived similarly.) $\bar{H}(\xi_1) = \bar{H}_2(\xi_1^{(2)}) + \bar{H}_3(\xi_1^{(3)}) + H_4(\xi_1^{(4)})$, where

$$\begin{aligned} \bar{H}_2(\xi_1^{(2)}) &= \frac{1}{2}\begin{bmatrix} Z^2_{T_1,\infty}(f(00), f(00+01)) & Z^2_{T_1,\infty}(f(01), f(01+01)) \\ Z^2_{T_1,\infty}(f(10), f(10+01)) & Z^2_{T_1,\infty}(f(11), f(11+01)) \end{bmatrix} \\ &= \frac{1}{2}\begin{bmatrix} Z^2_{T_1,\infty}(2E_s) & Z^2_{T_1,\infty}(2E_s) \\ Z^2_{T_1,\infty}(2E_s) & Z^2_{T_1,\infty}(2E_s) \end{bmatrix}. \end{aligned}$$

Here, E_s is the symbol energy; here also we have used the (somewhat abusive) notation, $Z_{A,B}(a) \triangleq Z_{A,B}(a, 0)$ for any real number a.

The fact that all the entries of $\bar{H}_2(\xi_1^{(2)})$ are identical suggests that the transfer function can be computed with scalar branch gains – which is indeed the case for any QPSK code.

All of the entries of $\bar{H}_3(\xi_1^{(3)})$ and $\bar{H}_4(\xi_1^{(4)})$ are similarly identical. The $(0,0)^{\text{th}}$ entry of $\bar{H}_3(\xi_1^{(3)})$ is given by:

$$[\bar{H}_3(\xi_1^{(3)})]_{0,0} = \frac{1}{2}\sum_{\ell=1}^{3}\prod_{j=1}^{3} Z_{A_{\ell,j},B_{\ell,j}}(f(\mathbf{c}_j), f(\mathbf{c}_j+\mathbf{e}_j))$$

where $(\mathbf{c}_1,\mathbf{c}_2,\mathbf{c}_3) = (00,00,00)$, $(\mathbf{e}_1,\mathbf{e}_2,\mathbf{e}_3) = (01,01,01)$, and $(A_{\ell,j}, B_{\ell,j})$ is the j^{th} element of $\Theta_{3,\ell}$. Then,

$$\begin{aligned}[\bar{H}_3(\xi_1^{(3)})]_{0,0} &= \frac{1}{2}[Z_{T_1,\infty}(2E_s)Z_{T_2,T_1}(2E_s)Z_{T_2,\infty}(2E_s) + Z_{T_2,T_1}(2E_s)\, Z_{T_1,\infty}(2E_s)Z_{T_2,\infty}(2E_s) \\ &\quad + Z^2_{T_2,T_1}(2E_s)Z_{T_2,\infty}(2E_s)]\end{aligned}$$

Similarly, each element of $\bar{H}_4(\xi_1^{(4)})$ is given by

$$\begin{aligned}[\bar{H}_4(\xi_1^{(4)})]_{0,0} &= \frac{1}{2}[Z_{0,T_2}(2E_s)Z^3_{0,\infty}(2E_s) + Z_{T_2,\infty}(2E_s)Z_{0,T_2}(2E_s)Z^2_{0,\infty}(2E_s) \\ &\quad + Z_{T_2,T_1}(2E_s)Z_{T_2,\infty}(2E_s)Z_{0,T_2}(2E_s)Z_{0,\infty}(2E_s) \\ &\quad + Z_{T_1,\infty}(2E_s)Z_{T_2,T_1}(2E_s)Z_{0,T_2}(2E_s)Z_{0,\infty}(2E_s)]\end{aligned}$$

3.2.3 Determination of Thresholds

Given three rate-compatible trellis codes of multiplicity two, three and four with rates R_2, R_3 and R_4 respectively, our goal is to determine the thresholds T_1 and T_2 that minimize the BER while maintaining a specified average rate R_{avg} $(R_4 \le R_{\text{avg}} \le R_2)$.

Let P_i be the probability that i symbols are sent during a given branch. Then

$$\begin{aligned}P_2 &= P(\rho_1 > T_1)\,P(\rho_2 > T_1) = \Big(\int_{T_1}^{\infty} 2\rho\exp(-\rho^2)\,d\rho\Big)^2 = e^{-2T_1^2}. \\ P_3 &= P(\rho_1 > T_1)\,P(T_2 < \rho_2 < T_1)\,P(\rho_3 > T_2) + P(T_2 < \rho_1 < T_1)\,P(\rho_2 > T_1)\,P(\rho_3 > T_2) \\ &\quad + P(T_2 < \rho_1 < T_1)\,P(T_2 < \rho_2 < T_1)\,P(\rho_3 > T_2) \\ &= e^{-T_2^2}(e^{-2T_2^2} - e^{-2T_1^2}) \\ P_4 &= 1 - P_2 - P_3.\end{aligned}$$

The average throughput is then specified by

$$\frac{P_2}{R_2} + \frac{P_3}{R_3} + \frac{P_4}{R_4} = \frac{1}{R_{\text{avg}}}.$$

This equation has infinite solutions in T_1 and T_2. We therefore select several valid values, plot the BER upper bound for each choice, and pick the best one.

Consider now the mother code of Example 3. For this code $R_2 = 1.5$ bit/symbol, $R_3 = 1.0$ bit/symbol and $R_4 = 0.75$ bit/symbol and the throughput equation is then,

$$\frac{e^{-2T_1^2}}{1.5} + \frac{e^{-T_2^2}(e^{-2T_2^2} - e^{-2T_1^2})}{1.0} + \frac{1 - e^{-2T_1^2} - e^{-T_2^2}(e^{-2T_2^2} - e^{-2T_1^2})}{0.75} = \frac{1}{R_{\text{avg}}}.$$

Suppose the desired average rate is $R_{\text{avg}} = 1$ bit/symbol. Then the above simplifies to

$$\frac{1}{3}e^{-2T_1^2}(e^{-T_2^2} - e^{-T_1^2}) - \frac{2}{3}e^{-2T_1^2} = \frac{1}{3}$$

In Table 4(a), we give some values of T_1 and T_2 that satisfy this equality, and the resulting probability of using each nested code on a given branch.

The upper bound on bit error-rate for each choice is depicted in Figure 7(a). The same analysis is repeated for $R_{\text{avg}} = 1.25$ bit/symbol in Table 4(b) and Figure 7(b).

3.2.4 Simulation Results

In this section, simulation is used to check the tightness of the bounds from Section 3.2.3 and to assess the adaptive codes' coding gain over some fixed rate codes from [10]. The trellis diagrams from [10] are given in Figure 8; for both codes, $L = 2$ and $d_p^2 = 16$.

Figure 7(a) and 7(b) show the upper bounds for each of the adaptive codes. Figure 7(c) compares the rate $R = 1$ bit/symbol adaptive codes; simulations indicate that the best adaptive code – Code 1, transmitting two, three, or four symbols with equal probability – out-performs the worst with a coding gain of about 4 dB at a BER of 10^{-5}. Figure 7(d) compares the 1 bit/symbol adaptive Code 1 with a fixed-rate code from [10]; the resulting coding gain is about 7 dB. In Figures 7(e) and 7(f), we provide analogous results for rate $R = 1.25$ bit/symbol codes. The gains of the best adaptive code (Code 1) over the worst adaptive code (Code 5) and over a fixed-rate 1.25 bits/symbol code from [10] are about 2 dB and 8 dB respectively.

3.3 Limiting the Variability of the Code Rate

The above protocol places no restriction on the number of punctured symbols in a frame; thus some buffering capability is necessary.

Consider the scheme in Example 1. Recall that this code represents five information bits per branch with either two or three channel symbols, and the targeted average rate is $R_{\text{avg}} = 2$ bits/symbol. If we want a *fixed* rate for each frame, then we would puncture exactly 125 symbols out of 750 – i.e., 250 branches (representing 1250 information bits) would be conveyed, with half the branches represented with two symbols and half with three symbols. If, however, we puncture symbols based on the algorithm in Section 3.1, the number of punctured symbols can vary from less than 40 to more than 200; Figure 9 is a histogram indicating the relative frequency of punctured symbols when variable puncturing is used.

To limit this variability, we alter the protocol so that exactly 125 symbols are punctured in each frame. The transmitter first sends 500 symbols of the frame – i.e., the first two symbols from each branch. The remaining 125 transmitted symbols are the third symbols from those branches most corrupted by the fading. Note that the thresholds of the previous protocol play no role here.

Figure 10 compares this fixed-rate approach with the variable-rate approach of Section 3.1. We see that the cost of eliminating the buffer variability is less than 0.5 dB at a BER of 10^{-5}.

4 Summary

This paper applied the notion of rate-compatibility to multiple TCM codes to construct "nested" trellis codes for channels with flat, slow Rayleigh fading. It showed that the codes in our rate-compatible families are at least as good as (and in some cases better than) corresponding codes from the literature. We then described a protocol that uses these codes in an adaptive hybrid-ARQ scheme with a low capacity feedback channel. We developed upper bounds on the performance of such a scheme and used these bounds to select the best code from several possibilities with the same average rate. Simulation results against fixed-rate codes from the literature demonstrated a substantial coding gain with no sacrifice of throughput. Finally, we described how to limit the variability of the code rate of such a scheme, thereby minimizing potential buffer problems. Simulation results in a particular case indicate that the loss incurred as a result of this rate-fixing is less than 0.5 dB.

References

[1] C. Schlegel and D. J. Costello, Jr., "Bandwidth Efficient Coding for Fading Channels: Code Construction and Performance Analysis", *IEEE Jnal. on Select. Areas in Comm.*, vol SAC-7. pp. 1356-1368, Dec. 1989.

[2] D. Divsalar and M. K. Simon, "The Design of Trellis Coded MPSK for Fading Channels: Set Partitioning for Optimum Code Design", *IEEE Trans. on Comm.,* vol. COM-36, pp. 1013-1021, Sep. 1988.

[3] E. Biglieri, D. Divsalar, P.J. McLane, M.K. Simon, *Introduction to Trellis-Coded Modulation with Applications*, Macmillan, New York, N.Y., 1991

[4] P. Ho, J. Cavers and J. Varaldi, "The Effects of Constellation Density on Trellis-Coded Modulation on Fading Channels", *IEEE Trans. on Veh. Tech.*, Aug. 1993, pp. 318-325

[5] Y. Liu and S. Blostein, "Identification of Frequency Nonselective Fading Channels using Decision Feedback and Adaptive Linear Prediction", *IEEE Trans. on Comm.*, vol. COM-43, pp. 1484-1492, Feb/Mar/Apr. 1995.

[6] D. Mandelbaum, "An Adaptive-Feedback Coding Scheme Using Incremental Redundancy", *IEEE Trans. Info. Theory*, vol. IT-20, pp. 388-389, May 1974.

[7] S. M. Alamouti, S. Kallel, "Adaptive Trellis-Coded Multiple-Phase-Shift Keying for Rayleigh Fading Channels", *IEEE Trans. on Comm.,* pp. 2305-2314, Jun. 1994.

[8] J. Hagenauer, "Rate Compatible Punctured Convolutional (RCPC) Codes and Their Applications", *IEEE Trans. on Comm.*, vol COM-36, pp. 389-400, Apr. 1988.

[9] J. B. Cain, G. C. Clark and J. M. Geist, “Punctured Codes of Rate (n-1)/n and Simplified Maximum Likelihood Decoding”, *IEEE Trans. Info. Theory*, pp. 97-100, Jan. 1979.

[10] E. Leonardo, L. Zhang, B. Vucetic, “Multidimensional M-PSK Trellis Codes for Fading Channels”, submitted to *IEEE Trans. Info. Theory*

[11] S. S. Periyalwar, S. M. Fleisher, “A Modified Design of Trellis-Coded MPSK for the Fading Channel”, *IEEE Trans. on Comm.*, vol. COM-41, pp. 874-882, Jun. 1993.

[12] M. Eröz and T. Fuja, “A Multiple Trellis-Coded Hybrid-ARQ Scheme for Land Mobile Communication Channels,” *MILCOM '95 Conference Record,* pp. 496-500.

M	$(n_1^\star)$
2	(1)
4	(1)
8	(3)
16	(7)
32	(7), (9)
64	(19), (27)

(a.) $k = 2$

M	$(n_1^\star\ n_2^\star)$
2	(1 1)
4	(1 1)
8	(1 3), (3 3)
16	(3 5), (3 7), (5 7)
32	(7 9), (7 15), (9 15)
64	(11 27), (15 29), (17 19)

(b.) $k = 3$

M	$(n_1^\star\ n_2^\star\ n_3^\star)$
2	(1 1 1)
4	(1 1 1)
8	(1 3 3)
16	(3 5 7)
32	(7 9 15)
64	(11 17 19), (11 23 27), (15 27 29), (19 25 29)

(c.) $k = 4$

M	$(n_1^\star\ n_2^\star\ n_3^\star\ n_4^\star)$
2	(1 1 1 1)
4	(1 1 1 1)
8	(1 1 3 3), (1 3 3 3)
16	(1 3 7 7), (1 5 7 7), (3 3 5 5)
32	(3 5 11 15), (3 7 13 15), (3 9 13 15), (5 7 9 11), (5 11 13 15)
64	(3 23 25 27), (5 17 19 23), (7 9 15 31), (7 9 17 23, (7 11 19 27), (7 15 23 25),(9 13 21 29), (9 15 17 25), (11 15 25 29)

(d.) $k = 5$

Table 1: Optimal multiplicative constants for $k = 2, 3, 4, 5$.

No. of States	Mod.	Bits/ Branch	Mult. Constants	Additive Constants	Throughput (bits/symb.)	L $(= k)$	d_p^2	d_{free}^2
2	4-PSK	3	1 1 1	000 213 123 332	1.5	2	4	4
					1.0	3	8	6
					0.75	4	16	8
4	4-PSK	3	1 1 1	000 213 132 301	1.5	2	8	6
					1.0	3	64	10
					0.75	4	256	16
2	8-PSK	4	3 1 3	000 642 246 404	2.0	2	2	2.34
					1.33	3	1.17	4.59
					1.0	4	4	8
4	8-PSK	5	3 1 3	000 274 421 615	2.5	2	1.17	1.76
				347 533 760 154	1.67	3	1.17	4.34
					1.25	4	4	6.93
8	8-PSK	5	3 1 3	000 612 271 427	2.5	2	1.17	1.76
				173 364 705 512	1.67	3	8	2.93
					1.25	4	16	6.34

Table 2: Some rate-compatible trellis codes

			Our codes				Other codes				
States	Throughput	Multiplicity	Const.	L	d_p^2	d_{free}^2	Const.	L	d_p^2	d_{free}^2	
2	2 bits/symbol	2	8-PSK	2	2	2.343	8-PSK	2	0.343		[10]
4	2.5 bits/symbol	2	8-PSK	2	1.172	1.757	8-PSK	2	0.343		[10]
2	1 bit/symbol	4	8-PSK	4	4	8	8-PSK	4	4	4.686	[2]
2	1 bit/symbol	4	8-PSK	4	4	8	16-PSK	4	4	8	[11]
2	1.5 bit/symbol	2	4-PSK	2	4	4	8-PSK	2	4	4	[11]

Table 3: Comparison of some nested codes from Table 2 against fixed-rate codes.

Code number	P_2	P_3	P_4	T_1	T_2
1	1/3	1/3	1/3	0.74	0.40
2	1/2	0	1/2	0.59	0.59
3	1/4	1/2	1/4	0.83	0.32
4	1/5	3/5	1/5	0.90	0.30
5	0.1	0.8	0.1	1.07	0.20
6	0	1	0	∞	0

(a)

Code number	P_2	P_3	P_4	T_1	T_2
1	0.75	0.1	0.15	0.38	0.28
2	0.7	0.2	0.1	0.42	0.21
3	0.8	0	0.2	0.33	0.33
4	0.65	0.3	0.05	0.46	0.15
5	0.6	0.4	0	0.51	0

(b)

Table 4: Some thresholds that achieve (a) $R_{\text{avg}} = 1$ bit/symbol (b) $R_{\text{avg}} = 1.25$ bit/symbol.

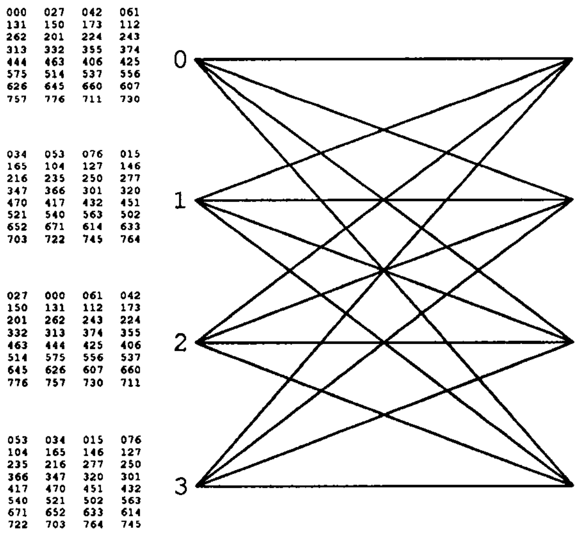


Figure 1: Trellis diagram of a mother code.

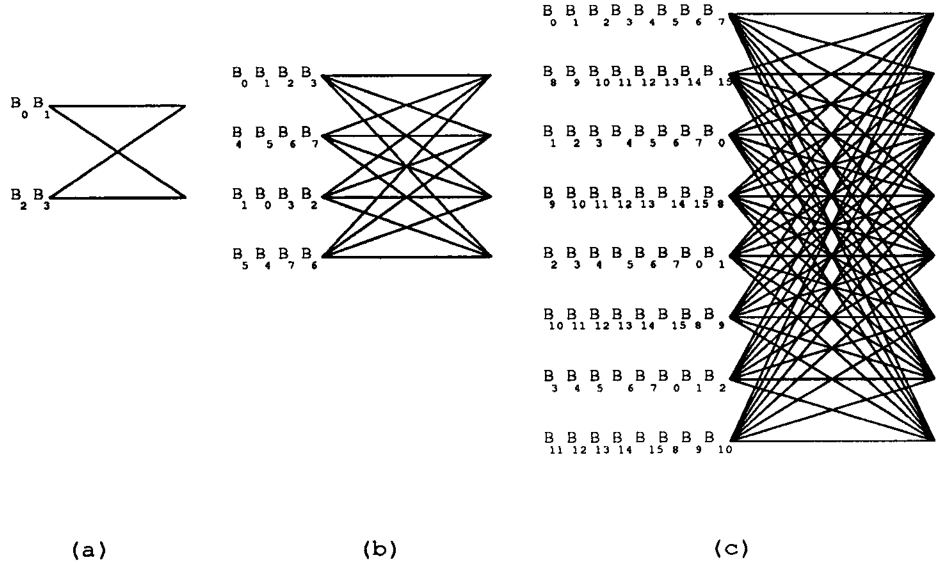

$B_0 B_1$
$B_2 B_3$
$B_0 B_1 B_2 B_3$
$B_4 B_5 B_6 B_7$
$B_1 B_0 B_3 B_2$
$B_5 B_4 B_7 B_6$
$B_0 B_1 B_2 B_3 B_4 B_5 B_6 B_7$
$B_8 B_9 B_{10} B_{11} B_{12} B_{13} B_{14} B_{15}$
$B_1 B_2 B_3 B_4 B_5 B_6 B_7 B_0$
$B_9 B_{10} B_{11} B_{12} B_{13} B_{14} B_{15} B_8$
$B_2 B_3 B_4 B_5 B_6 B_7 B_0 B_1$
$B_{10} B_{11} B_{12} B_{13} B_{14} B_{15} B_8 B_9$
$B_3 B_4 B_5 B_6 B_7 B_0 B_1 B_2$
$B_{11} B_{12} B_{13} B_{14} B_{15} B_8 B_9 B_{10}$
(a)
(b)
(c)

Figure 2: Trellis structures (a) two-state code (b) four-state code (c) eight-state code.

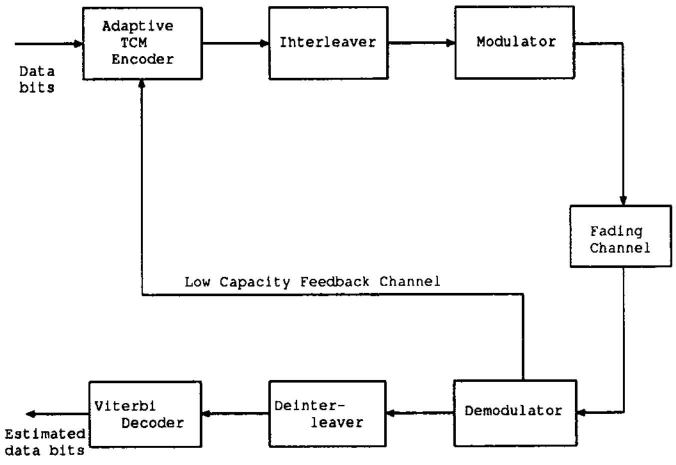


Figure 3: Block diagram of a communication system with feedback.

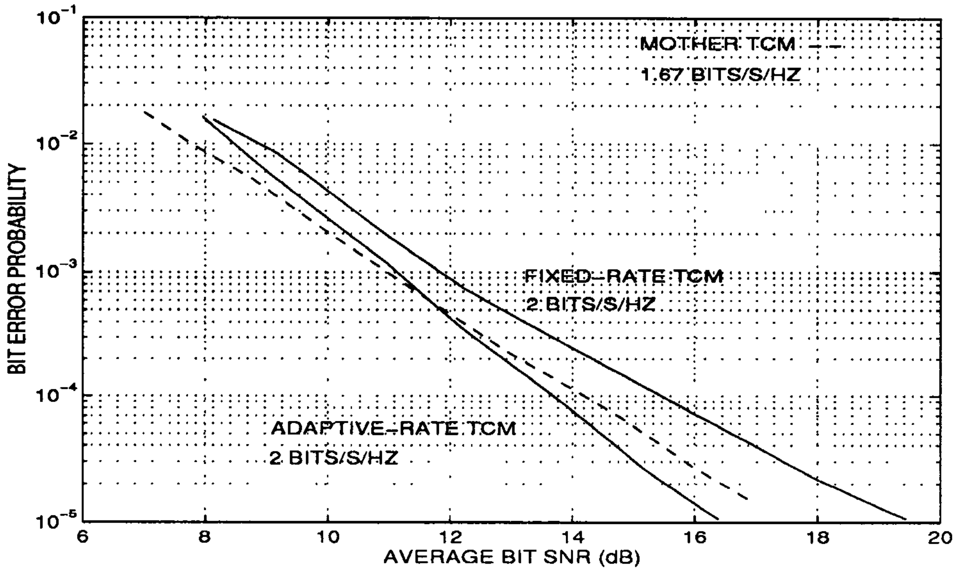


Figure 4: Bit error rate performance of adaptive and fixed-rate codes.

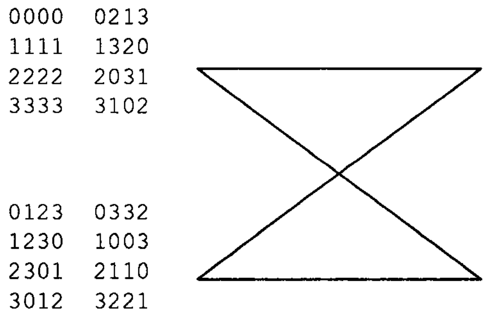
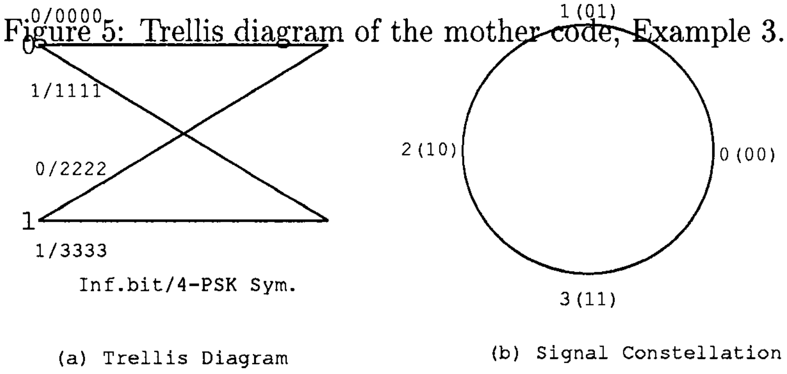


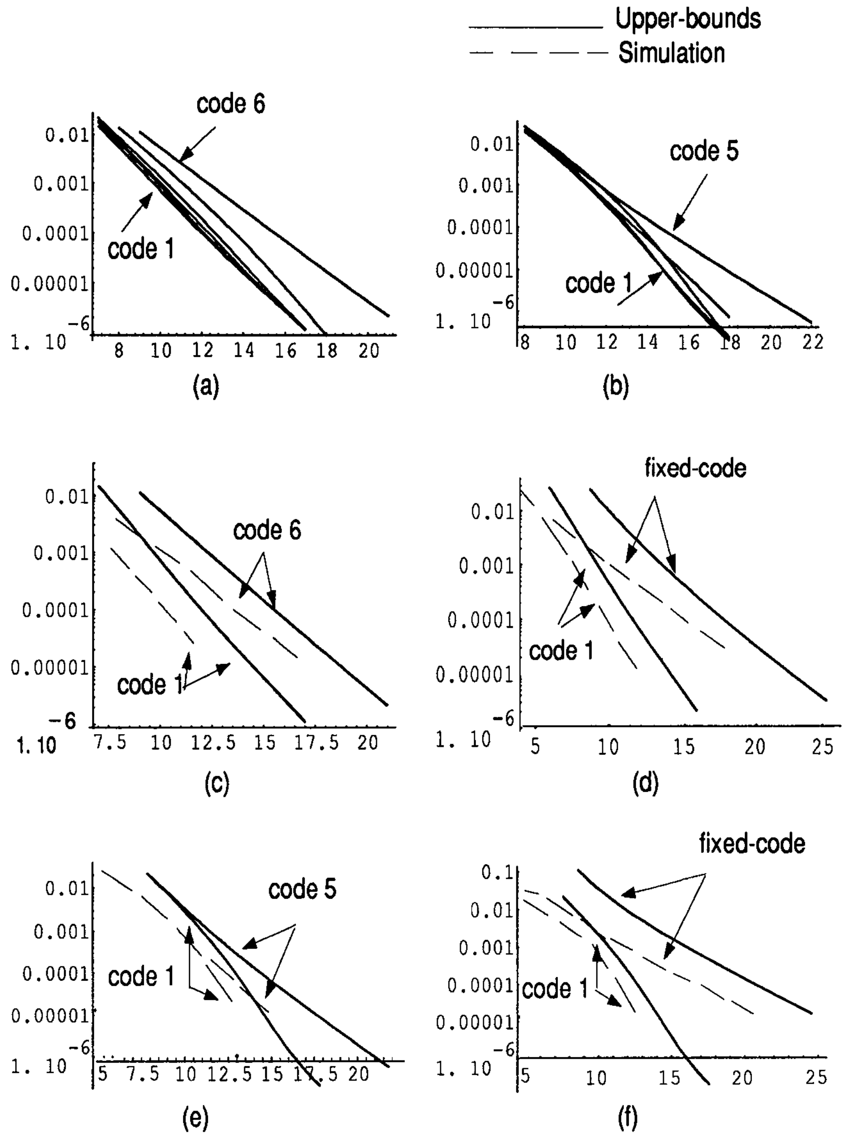
Figure 5: Trellis diagram of the mother code, Example 3.



(a) Trellis Diagram

(b) Signal Constellation

Figure 6: Trellis diagram of the mother code and signal constellation, Example 4.



(a) R=1 bit/s/hz, Upper bounds on adaptive codes
(b) R=1.25 bit/s/Hz, Upper bounds on adaptive codes
(c) R=1 bit/s/Hz, Best vs worst thresholds
(d) R=1 bit/s/Hz, Best threshold vs fixed-rate code
(e) R=1.25 bit/s/Hz, Best vs worst thresholds
(f) R=1.25 bit/s/Hz, Best threshold vs fixed-rate code

Figure 7: y-axis: Bit error rate, x-axis: Average bit SNR

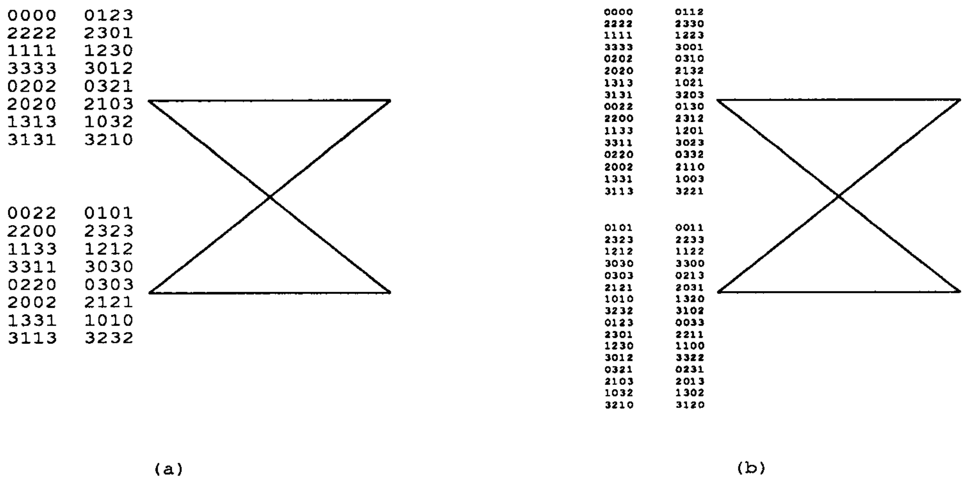


Figure 8: Trellis diagram of a fixed-rate code, $R = 1$ bit/symbol [10]

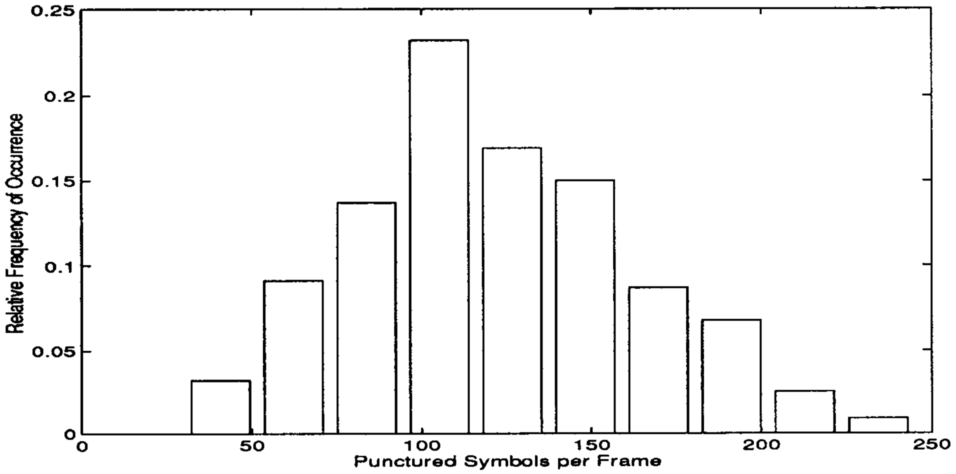


Figure 9: Histogram analysis of Example 1

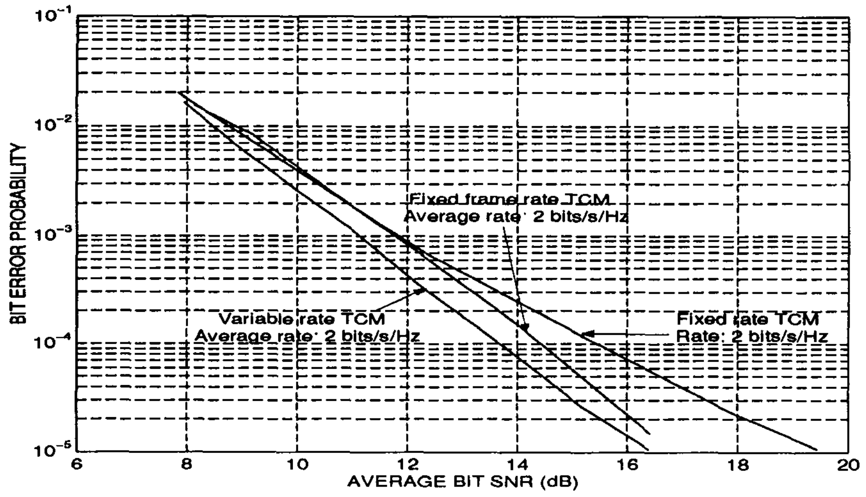


Figure 10: Comparison of bit error rate performances including "fixed"-rate modified system