Symbol-by-Symbol MAP Demodulation of CPM and PSK Signals on Rayleigh Flat-Fading Channels

Michael J. Gertsman, *Member, IEEE*, and John H. Lodge, *Senior Member, IEEE*

***Abstract*— Demodulation using the symbol-by-symbol maximum *a posteriori* probability (MAP) algorithm is presented. The algorithm is derived for the case of continuous phase modulation (CPM) signals transmitted over Rayleigh flat-fading channels, and a corresponding receiver structure is specified. It is shown that the MAP algorithm requires computing, for each trellis branch, the sum of the products of the weights of all paths through the trellis which pass through that branch, and that this generic computational problem can be solved efficiently by an approach that uses a forward and backward recursion through the trellis. Simulation results are presented which show both the hard and soft decision performance of the MAP receiver to be robust, even in the presence of fade rates of up to 30% of the symbol rate. The application of the receiver concept to phase-shift keying (PSK) signals is also discussed, and then evaluated via simulation. The concept of joint demodulation and decoding using iterative processing techniques is introduced. It is shown that the MAP receiver is well suited for iterative processing applications due to its use of *a priori* symbol probabilities and its production of optimal soft decisions. Simulation results for the reception of quaternary PSK (QPSK) show that the bit error rate (BER) performance of the iterative MAP receiver can approach that of a receiver operating with perfect knowledge of the fading process.**

***Index Terms*— Iterative decoding, MAP detection, Rayleigh channels.**

I. Introduction

INTEREST in terrestrial mobile communications is expanding rapidly. While studies vary considerably in their predictions, it is clear that contention for spectrum allocation will necessitate the increasingly efficient usage of bandwidth [1]. Cost, size, and interference considerations will place similar requirements on signal power. To meet these and other challenges, digital transmission has emerged as the technique of choice due to its robustness and more efficient power and bandwidth utilization [2], [3].

Bandwidth limitations have motivated considerable investigation into continuous phase modulation (CPM) techniques [4]–[6]. CPM signals are of interest due to the spectral efficiency which they can achieve, coupled with their constant envelope property. These properties are particularly attractive for mobile communications systems, which are interference limited (demanding stringent adjacent channel interference specifications) and cost driven, since nonlinear amplifiers may be used without producing spectral regrowth.

Bandwidth limitations also have motivated the trend toward the use of higher frequency bands. While current mobile communications systems operate at frequencies at L band (1–2 GHz) or below, systems utilizing spectrum at up to Ka band (27–40 GHz) are being investigated [7]. A significant amount of spectrum is allocated for mobile communications services in these higher frequency bands.

Commensurate with the higher frequencies are higher fading rates. Fading rates are proportional to both vehicle speed and carrier frequency; therefore, future systems could experience fading rates one or two orders of magnitude higher than those found at L band. In this paper, we wish to consider fast-fading channels, which are defined as channels where the phase of the fading process can vary significantly over a symbol period, i.e., where the symbol duration is a significant fraction of the coherence time of the channel [8, p. 715]. This corresponds to channels where the single-sided fading bandwidth is a significant proportion of the symbol rate, with fade rates as high as 30% of the symbol rate being considered, herein.

Terrestrial mobile communications channels often are modeled as Rayleigh flat-fading channels [9]. Detection of signals on such channels traditionally has been performed in a noncoherent or differentially coherent manner. The techniques utilized exhibit irreducible error floors at high fade rates, generate suboptimal soft decisions leading to power inefficiency, or in the case of pilot symbol approaches [10], [11] on fast-fading channels, require greater bandwidth.

In [12], a receiver structure for fading channels is developed which uses maximum-likelihood sequence estimation (MLSE) to determine the data sequence which produces the most likely channel samples. This receiver overcomes both the error floor limitation of differential detection and the power/bandwidth limitation of pilot symbol approaches. Its major drawback is that it produces hard decisions via the Viterbi algorithm. This makes it unsuitable for applications requiring soft decisions in subsequent stages. This limitation is particularly severe given the trend in digital transmission systems toward concatenated processing structures [13] (e.g., modems utilizing forward error-correction codes with interleaving) where performing joint demodulation and decoding in a single stage is impractical due to the number of states required.

As is well known, the symbol-by-symbol maximum *a posteriori* probability (MAP) algorithm is the optimum decoding

Paper approved by A. Goldsmith, the Editor for Wireless Communication of the IEEE Communications Society. Manuscript received May 27, 1996; revised September 15, 1996 and December 16, 1996. This work was supported in part by the Spectrum Research Funds of Industry Canada. This paper was presented in part at Wireless'95, Calgary, Alta., Canada, July 1995.

M. J. Gertsman is with Square Peg Communications Inc., Ottawa, Ont., K2H 8S2 Canada (e-mail: michael.gertsman.squarepeg@crc.doc.ca).

J. H. Lodge is with the Communications Research Centre, Ottawa, Ont., K2H 8S2 Canada.

Publisher Item Identifier S 0090-6778(97)05185-4.

algorithm for codes that can be represented by a trellis of finite duration [13]–[15]. The algorithm is of interest because it produces, by definition, optimal soft decisions (decisions which have the maximum *a posteriori* probability). This paper builds on the work in [12] by examining the use of the MAP algorithm for demodulation.

Since symbol-by-symbol MAP demodulation is a soft-in/soft-out process which utilizes *a priori* symbol probabilities at its input and produces optimal soft decisions at its output, the algorithm is well suited to iterative processing applications, where refined input symbol probabilities are fed back to the demodulator as *a priori* information. The resulting refined soft decisions produce corresponding improvements in the subsequent processing stages. This approach has demonstrated significant potential for decoding applications [16]–[18], but has not been investigated previously for combined demodulation/decoding applications.

In the remainder of this paper, we

- derive the symbol-by-symbol MAP algorithm for the demodulation of CPM signals on Rayleigh flat-fading channels;
- determine the performance of the symbol-by-symbol MAP algorithm relative to that of alternate detection schemes;
- evaluate the effectiveness of the symbol-by-symbol MAP algorithm in providing soft decisions for a following deinterleaver/convolutional decoder;
- evaluate the use of iterative MAP processing techniques for joint demodulation and decoding.

To accomplish these goals, a combination of theory, analysis, and Monte Carlo simulation is used. Performance is evaluated both for a binary CPM scheme and for QPSK, which is considered to be a discrete-time constant envelope modulation.

II. Symbol-by-Symbol MAP Demodulation

In this section, the symbol-by-symbol MAP algorithm is applied to the demodulation of CPM signals transmitted over Rayleigh flat-fading channels.

A. Approach

Given a particular sequence of received samples, the goal of symbol-by-symbol MAP demodulation is to determine, for all possible symbols and all possible times t, the probability that a particular symbol was transmitted at that time. Once these probabilities have been determined, the demodulator can use them directly as symbol soft decisions, or can process them further to extract bit soft decisions.

The development of symbol-by-symbol MAP demodulation is presented in four stages. First, it is shown that the problem of determining the state transition probabilities can be thought of in terms of a generic computational problem over a trellis with multiplicative branch weights. Second, an efficient technique for the required computations is described. Third, obtaining the soft decisions using state transition probabilities is discussed. These first three stages are general in nature, but assume that the demodulator can compute the probability density of the present received signal sample, conditioned upon the past received signal samples and the subset of hypothesized input symbols that form the state vector. The final stage describes how the demodulator can obtain the required conditional probability density values for the special case of CPM and PSK signals on Rayleigh flat-fading channels.

B. State Transition Probabilities

The following notational definitions will be applied.

Q $\triangleq$ number of data symbols (Q-ary modulation);
N $\triangleq$ length of data block, in symbol periods;
M $\triangleq$ number of states in trellis at a given time;
k $\triangleq$ hypothesis;
r $\triangleq$ number of samples/symbol;
t $\triangleq$ discrete-time index, in symbol periods;
$\tilde{\boldsymbol{U}}$ $\triangleq N$—vector of transmitted symbols;
$\tilde{\boldsymbol{X}}$ $\triangleq rN$—vector of transmitted symbol samples;
$\boldsymbol{Y}$ $\triangleq rN$—vector of received symbol samples;
$\boldsymbol{U}(k)$ $\triangleq N$—vector of Q-ary input symbols for hypothesis k;
$\boldsymbol{X}(k)$ $\triangleq rN$—vector of transmitted samples for hypothesis k.

Here, it is assumed that the combined memory of the modulator and the channel is no greater than K symbol periods, so that any given received sample is affected by no more than $K+1$ consecutive input symbols. Consequently, a trellis with no more than $M = Q^K$ states, at the start of each symbol interval, can be used. Each node in the trellis has Q input branches and Q output branches, with a branch corresponding to one of the Q data symbols. It also is assumed that we are operating on a block of data that starts and ends in a known state. One way to achieve this condition is to start and end each data block with a "unique" word (i.e., known symbols), where the length of the unique word is at least K symbols.

The state transition probabilities are intermediate quantities of great interest because they can be used to compute the probability that a given symbol was transmitted at time t. Define the set $C_t(m', m)$ as the subset of the set of hypotheses $\{k\}$ that traverse the trellis branch between states $S_{t-1} = m'$ and $S_t = m$. The state transition probability is given by

$$\Pr\{S_{t-1} = m'; S_t = m|\boldsymbol{Y}\} = \frac{\displaystyle\sum_{k \in C_t(m', m)} \Pr\{\tilde{\boldsymbol{U}} = \boldsymbol{U}(k)|\boldsymbol{Y}\}}{\displaystyle\sum_{k} \Pr\{\tilde{\boldsymbol{U}} = \boldsymbol{U}(k)|\boldsymbol{Y}\}}. \tag{1}$$

To compute (1), we need to compute $\Pr\{\tilde{\boldsymbol{U}} = \boldsymbol{U}(k)|\boldsymbol{Y}\}$. Using Bayes' theorem (e.g., [19, eq. (7-45)]), this can be written

$$\Pr\{\tilde{\boldsymbol{U}} = \boldsymbol{U}(k)|\boldsymbol{Y}\} = \frac{p(\boldsymbol{Y}|\tilde{\boldsymbol{U}} = \boldsymbol{U}(k)) \cdot \Pr\{\tilde{\boldsymbol{U}} = \boldsymbol{U}(k)\}}{p(\boldsymbol{Y})}. \tag{2}$$

When (2) is substituted into (1), all terms will have the identical denominator [i.e., $p(\boldsymbol{Y})$], which will cancel, and therefore, can be ignored.

In evaluating the numerator of (2), we make the assumption that the Q-ary input symbols, transmitted at distinct times t, are independent. Therefore, the second factor of the numerator becomes

$$\Pr\{\tilde{\boldsymbol{U}} = \boldsymbol{U}(k)\} = \prod_{t=0}^{N-1} \Pr\{\tilde{u}_t = u_t(k)\}. \tag{3}$$

Now, consider the first factor in the numerator $p(\boldsymbol{Y}|\tilde{\boldsymbol{U}} = \boldsymbol{U}(k))$. Note that there is a 1-to-1 correspondence between the input vector $\boldsymbol{U}(k)$ and the transmitted sample vector $\boldsymbol{X}(k)$. Consequently

$$p(\boldsymbol{Y}|\tilde{\boldsymbol{U}} = \boldsymbol{U}(k)) = p(\boldsymbol{Y}|\tilde{\boldsymbol{X}} = \boldsymbol{X}(k)). \tag{4}$$

From [12, eq. (3)], we know that the right-hand side of (4) is given by

$$\begin{aligned} p_{y|x}[\boldsymbol{Y}|\boldsymbol{X}(k)] &= p_{y|x}[(y_{rN-1}, \cdots, y_1, y_0)|\boldsymbol{X}(k)] \\ &= \prod_{j=0}^{rN-1} p_{y_j|\mathcal{Y}_{j-1}}[y_j|\mathcal{Y}_{j-1}(k)] \end{aligned} \tag{5}$$

where

$$\mathcal{Y}_j(k) = \{(y_j, \cdots, y_1, y_0), \boldsymbol{X}(k)\}.$$

The computation of $p_{y_j|\mathcal{Y}_{j-1}}[y_j|\mathcal{Y}_{j-1}(k)]$ is dependent upon the channel model and modulation scheme. For the special case of CPM over a Rayleigh flat-fading channel, the computation will be described in Section II-E. At this point, we assume only that it can be computed, and that the computation is dependent upon at most $K+1$ symbols corresponding to a trellis with Q^K states.

In the trellis formulation, for any given trellis branch, each possible symbol at a given time is uniquely specified by the starting and ending states of the corresponding branch at that time. Therefore, there is a 1-to-1 correspondence between the sequence of input symbols $\boldsymbol{U}(k) = \{u_0(k) \;\; u_1(k) \;\; \cdots \;\; u_{N-1}(k)\}$ and the sequence of state transitions $\boldsymbol{M}(k) = \{(m', m)_0 \;\; (m', m)_1 \;\; \cdots \;\; (m', m)_{N-1}\}$, where the $\{(m', m)_t\}$ represent the starting and ending states for the kth hypothesis at time t, and m' at time t is the same as m at time $t-1$. Define

$$W_t(m', m) = \prod_{j=rt}^{r(t+1)-1} p_{y_j|\mathcal{Y}_{j-1}}[y_j|\mathcal{Y}_{j-1}(k)]. \tag{6}$$

Substituting (6) into (5) yields

$$p_{y|x}[\boldsymbol{Y}|\boldsymbol{X}(k)] = \prod_{t=0}^{N-1} W_t(m', m) \tag{7}$$

where the $\{W_t(m', m)\}$ are not dependent upon the hypothesis k since the computation of $p_{y_j|\mathcal{Y}_{j-1}}[y_j|\mathcal{Y}_{j-1}(k)]$ is dependent upon at most $K+1$ consecutive symbols.

Using (3), (4), and (7), the numerator of (2) becomes

$$\begin{aligned} &p(\boldsymbol{Y}|\tilde{\boldsymbol{U}} = \boldsymbol{U}(k)) \cdot \Pr\{\tilde{\boldsymbol{U}} = \boldsymbol{U}(k)\} \\ &\quad = \prod_{t=0}^{N-1} \Pr\{\tilde{u}_t = u_t(k)\} \cdot W_t(m', m) \\ &\quad = \prod_{t=0}^{N-1} \gamma_t(m', m) \end{aligned} \tag{8}$$

where

$$\gamma_t(m', m) = \Pr\{\tilde{u}_t = u_t(k)\} \cdot W_t(m', m). \tag{9}$$

Again, while the sequence of multiplicative branch weights is a function of the hypothesis k, individual multiplicative branch weights $\gamma_t(m', m)$ are not functions of the hypothesis k because each branch is associated with a specific symbol. Consequently, all hypotheses that pass through a given branch will include the specific symbol in that time period.

In terms of the multiplicative branch weights, the state transition probability of (1) becomes

$$\Pr\{S_{t-1} = m'; S_t = m|\boldsymbol{Y}\} = \frac{\displaystyle\sum_{k \in C_t(m'_t, m_t)} \prod_{i=0}^{N-1} \gamma_i(M_i(k))}{\displaystyle\sum_k \prod_{i=0}^{N-1} \gamma_i(M_i(k))} \tag{10}$$

where $M_i(k)$ denotes the ith element of the hypothesized path through the trellis $M(k)$.

C. Efficient Computation of the State Transition Probabilities

The quantity on the right-hand side of (10) is recognized to be a generic computational problem. That is, for each branch in the trellis, we wish to compute the ratio

$$\frac{\text{sum of the products of the weights of all paths through the trellis which pass through the given branch}}{\text{sum of the products of the weights of all paths through the trellis}}.$$

Here, we briefly describe a computationally efficient approach to the above problem.

As an example, consider the length $N=7$ trellis shown in Fig. 1(a) where the nodes correspond to one of $M=2$ states and the branches represent possible transitions. Assume that we wish to compute the sum of the products of the weights of all paths through the trellis (there are 2^6 paths which start and end in state 0). An efficient way to perform this computation is via a recursion of the form

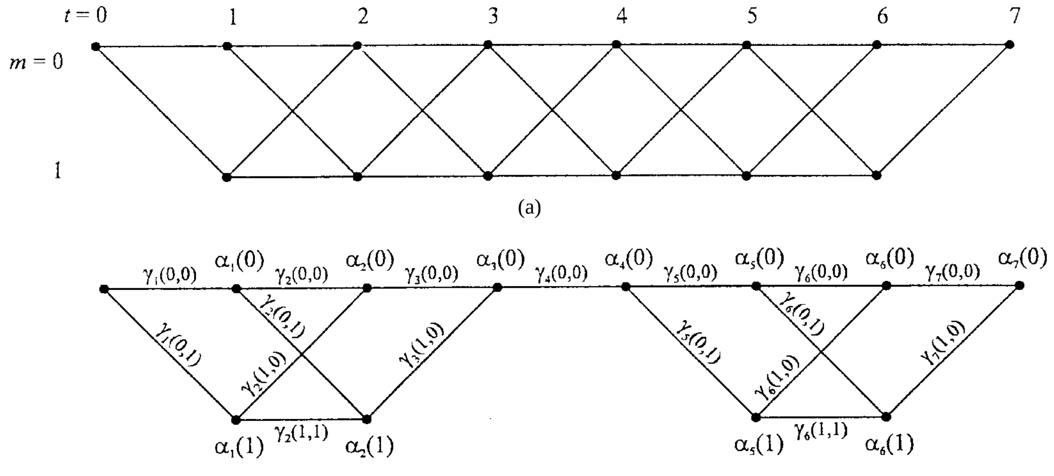
$$\alpha_t(m) = \sum_{m'} \alpha_{t-1}(m') \cdot \gamma_t(m', m) \tag{11}$$

where $\alpha_t(m)$ is the sum of the products of the weights along all paths which terminate in state m at time t and $\alpha_0(m)$, $0 \le m \le M$ are known initial conditions. The sum over m' is for all cases for which a branch exists from state m' to state m at time t, and $\gamma_t(m', m)$ is the corresponding branch weight.

Now, consider the case where we wish to solve for the sum of the products of the weights for only those paths which traverse a particular branch in the trellis. In this case, it is advantageous to split the trellis into two parts. This is shown in Fig. 1(b) for the case of the branch from $S_3 = 0$ to $S_4 = 0$. From (11), $\alpha_3(0)$ gives us the sum of the products of the weights along all subpaths which terminate in state 0 at time $t=3$. The weight on the branch itself is $\gamma_4(0, 0)$. To compute the summation for the second half of the trellis, a backward recursion can be used. This recursion has the form

$$\beta_t(m) = \sum_{m'} \beta_{t+1}(m') \cdot \gamma_{t+1}(m, m') \tag{12}$$

where $\beta_t(m)$ is the sum of the products of the weights along all



(a)

(b)

Fig. 1. (a) Trellis of the example Markov process. (b) Equivalent trellis when only those paths passing through the branch from state $S_3 = 0$ to state $S_4 = 0$ are considered.

paths to the end of the trellis which originate from state m at time t and $\beta_N(m)$, $0 \leq m \leq M$ are known final conditions. Here, the sum over m' is for all cases for which a branch exists from state m to state m' at time t, and $\gamma_t(m', m)$ is the corresponding branch weight.

In general, if we define $\sigma_t(m', m)$ as the sum of the products of the weights of all paths which pass through the branch delineated by $S_{t-1} = m'$ and $S_t = m$, then

$$\sigma_t(m', m) = \alpha_{t-1}(m') \cdot \gamma_t(m', m) \cdot \beta_t(m). \tag{13}$$

Note that to solve (13) for all branches of the trellis, the forward and backward recursions (over the whole trellis) only need to be performed once.

Equations (11)–(13) represent an efficient recursive approach for addressing the generic computational problem posed at the beginning of this subsection. The form of this approach is essentially that of the algorithm introduced by Bahl *et al.* [14], but differs from it in two important respects.

1) While the Bahl algorithm was derived for the specific case of probabilities governing a discrete-time finite-state Markov process, in this paper, we recognize that the algorithm in [14], and others like it, are simply efficient ways to solve the generic computation problem stated at the beginning of this subsection.
2) The definitions of $\alpha_t(m)$ and $\beta_t(m)$ presented herein are symmetric, while those in [14] are asymmetric.

D. Computation of Soft Decisions

For demodulation purposes, the quantity of interest is $\Pr\{u_t = q|\boldsymbol{Y}\}$ where q is one of the Q-ary input symbols. This quantity is calculated by summing the transition probabilities that correspond to branches associated with the symbol q at time t.

Define a set of all state transitions for which $u_t = q$:

$$A = \{(m', m) \colon u_t(m', m) = q\}$$

where $u_t(m', m)$ is the transmitted symbol corresponding to the branch from state $S_{t-1} = m'$ to state $S_t = m$. Then

$$\Pr\{u_t = q|\boldsymbol{Y}\} = \sum_{(m', m) \in A} \Pr\{S_{t-1} = m'; S_t = m|\boldsymbol{Y}\}. \tag{14}$$

Making use of the fact that the numerator of (10) is given by $\sigma_t(m', m)$, the desired probability becomes

$$\Pr\{u_t = q|\boldsymbol{Y}\} = \frac{\displaystyle\sum_{(m', m) \in A} \sigma_t(m', m)}{\displaystyle\sum_{(m', m)} \sigma_t(m', m)}. \tag{15}$$

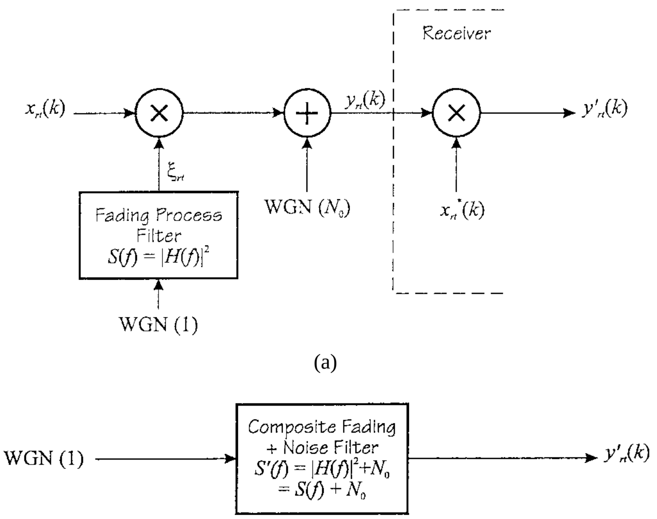
Note that the denominator of (15) is, in fact, the sum of the products of the weights of all paths through the trellis, and is therefore, the same at all times t. In fact, from the definition of $\alpha_t(m)$, (14) also can be written

$$\Pr\{u_t = q|\boldsymbol{Y}\} = \frac{\displaystyle\sum_{(m', m) \in A} \sigma_t(m', m)}{\displaystyle\sum_{m} \alpha_N(m)} \tag{16}$$

where the summation in the denominator is over the possible final states of the trellis.

E. The Special Case of CPM and PSK Signals on Rayleigh Flat-Fading Channels

From (11) to (13), we can see that the computation of the branch probabilities (and thus the symbol probabilities) revolves around the determination of the branch weights $\gamma_t(m', m)$ and the structure of the trellis. The trellis is straightforward since it is simply determined by the bit (or symbol) sequence which is affected by the memory of the channel. The value of $\gamma_t(m', m)$ is a function of the *a priori* symbol probabilities and the probability distribution function of the received channel samples conditioned on the input and the previously received samples, i.e., $p_{y_j|\mathcal{Y}_{j-1}}[y_j|\mathcal{Y}_{j-1}(m', m)]$.


Receiver
$x_{rt}(k)$
$y_{rt}(k)$
$y'_{rt}(k)$
ξ_{rt}
Fading Process Filter $S(f) = |H(f)|^2$
WGN (N_0)
$x^*_{rt}(k)$
WGN (1)
Composite Fading + Noise Filter $S'(f) = |H(f)|^2+N_0 = S(f) + N_0$
WGN (1)
$y'_{rt}(k)$

(a)

(b)

Fig. 2. Statistically equivalent transmission models (from [12]). WGN(σ) refers to a white Gaussian noise process with variance σ.

While the previous results of this section are generally applicable to the demodulation of signals in any system which has memory, we now restrict ourselves to the case of CPM signals transmitted over Rayleigh flat-fading channels. Consider a system such as that shown in Fig. 2(a), where $x_{rt}(k)$ represents the transmitted symbol sample for hypothesis k at time rt, $S(f)$ represents the power spectral density of the fading process filter, and $y_{rt}(k)$ represents the corresponding received symbol sample. If the receiver removes the modulation corresponding to hypothesis k, we are left with $y'_{rt}(k)$, which represents the combined effect of the multiplicative fading and the additive noise. Note that removing the modulation is a linear transformation, with a Jacobian of magnitude one for CPM; therefore,

$$p_{\boldsymbol{Y}'|\boldsymbol{X}}[\boldsymbol{Y}'(k)|\boldsymbol{X}(k)] = p_{\boldsymbol{Y}|\boldsymbol{X}}[\boldsymbol{Y}|\boldsymbol{X}(k)]. \tag{17}$$

In [12], it is shown that for constant envelope signals whose signal amplitude is 1 (for convenience only), $\boldsymbol{Y}'$ is a zero-mean complex Gaussian process with covariance $\boldsymbol{R}$ where $\boldsymbol{R}$ is the Toeplitz covariance matrix whose elements correspond to the autocorrelation coefficients of the composite fading plus noise power spectral density function $S'(f)$ shown in Fig. 2(b). The conditional pdf of $\boldsymbol{Y}'(k)$, the fading plus noise, given the input $\boldsymbol{X}(k)$ then can be written as

$$p_{\boldsymbol{Y}'|\boldsymbol{X}}[\boldsymbol{Y}'(k)|\boldsymbol{X}(k)] = [\pi^N|\boldsymbol{R}|]^{-1}e^{-\boldsymbol{Y}'(k)^H\boldsymbol{R}^{-1}\boldsymbol{Y}'(k)}$$

where $(\cdot)^H$ represents the conjugate transpose of a matrix or vector.

The representation shown in Fig. 2(b) suggests a receiver structure which uses a predictor to estimate the composite fading plus noise process. In [12, Theorem 1] it is shown how to compute $\boldsymbol{R}$ from the outputs of an all-pole linear predictor, given a hypothesized transmitted sequence.

It is shown in [12, Theorem 2] that

$$\begin{aligned} &p_{y_j|\mathcal{Y}_{j-1}}[y_j|\mathcal{Y}_{j-1}(k)] \\ &\quad= \frac{r_0}{\pi\sigma^2_{j|j-1}(k)}\exp\left[-\frac{|y'_j - y'_{j|j-1}(k)|^2}{\sigma^2_{j|j-1}(k)}\right] \end{aligned} \tag{18}$$

where

$$y'_{j|j-1}(k) = \sum_{i=1}^{j-1} a_i^{j-1}y'_{j-i}(k) \tag{19}$$

$$\sigma^2_{j|j-1}(k) = r_0V_{j-1} \tag{20}$$

a_i^{j-1} is the ith coefficient of the ($j-1$)th-order linear predictor, V_{j-1} is the corresponding expected normalized squared prediction error ($V_0 = 1$), and r_0 is the zero-lag autocorrelation coefficient.

If it is assumed that the composite fading plus noise filter can be approximated accurately by an all-pole model of order L, all linear predictors of order L and greater have essentially the same coefficients, and therefore the same expected normalized squared prediction error. Assuming a known preamble, the contribution of predictor orders less than L can be ignored. Therefore, using (18)–(20) and the fact that $V_{j-1} = V_L$ for $j > L$, we can rewrite (6) as

$$\begin{aligned} W_t(m', m) &= \prod_{j=rt}^{r(t+1)-1} \frac{r_0}{\pi\sigma^2_{j|j-1}(k)}\exp\left[-\frac{|y'_j - y'_{j|j-1}(k)|^2}{\sigma^2_{j|j-1}(k)}\right] \\ &= \prod_{j=rt}^{r(t+1)-1} \frac{1}{\pi V_L}\exp\left[-\frac{\left|\sum_{i=0}^{L} a_i^L x^*_{j-i}(k)y_{j-i}\right|^2}{r_0V_L}\right] \\ &= \left(\frac{1}{\pi V_L}\right)^r\exp\left[-\frac{\sum_{j=rt}^{r(t+1)-1} B_j(k)}{r_0V_L}\right] \end{aligned} \tag{21}$$

where

$$B_j(k) = \left|\sum_{i=0}^{L} a_i^L x^*_{j-i}(k)y_{j-i}\right|^2 \tag{22}$$

and $a_0^L = -1$. Note that the summation term in (22) represents the output of a prediction error filter for y' combined with the removal of the hypothesized modulation over a sequence of L received samples.

Since the computation of the soft decision (15) is a ratio of sums, with each term composed of the product of N factors of the form of (21), the common factor $(\pi V_L)^{-r}$ can be omitted, yielding

$$W'_t(m', m) = \exp\left[-\frac{\sum_{j=rt}^{r(t+1)-1} B_j(k)}{r_0V_L}\right]. \tag{23}$$

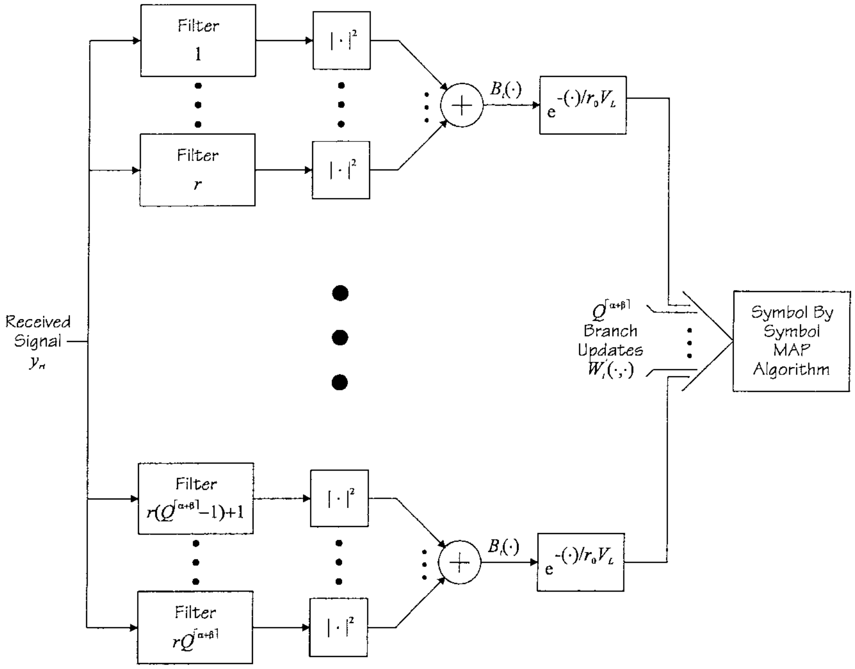

Filter 1
$|\cdot|^2$
Filter r
$|\cdot|^2$
$B_t(\cdot)$
$e^{-(\cdot)/r_0V_L}$
Received Signal y_n
$Q^{\lceil\alpha+\beta\rceil}$ Branch Updates $W_t'(\cdot,\cdot)$
Symbol By Symbol MAP Algorithm
Filter $r(Q^{\lceil\alpha+\beta\rceil}-1)+1$
$|\cdot|^2$
Filter $rQ^{\lceil\alpha+\beta\rceil}$
$|\cdot|^2$
$B_t(\cdot)$
$e^{-(\cdot)/r_0V_L}$

Fig. 3. Filter bank structure used by the MAP demodulator.

Then the MAP processing used to produce soft decisions employs $W_t'(m', m)$ in place of $W_t(m', m)$ in (9), and then uses (11)–(13) and (15).

The calculation of the values for $W_t'(m', m)$ can be implemented using the matched filter bank shown in Fig. 3. Each filter combines the removal of the hypothesized modulation with the prediction error filter function required for the computation of $B_t(k)$. The form and length of these filters are derived in [12]. It is shown that all calculations required for the computation of the $B_t(k)$'s for a single symbol period are affected by no more than $\lceil\alpha + \beta\rceil$ symbols where α represents the pulse length in symbol periods and β represents the predictor length in symbol periods. Since this span of symbols represents the memory required in the demodulator, the number of states required for Q-ary modulation is given by $Q^{\lceil\alpha+\beta\rceil-1}$.

Referring again to Fig. 3, the outputs of the filters for a given symbol period and hypothesis are combined (if the number of samples/symbol is greater than 1), the result is divided by the expected prediction error r_0V_L, and then the exponent is taken.

A few points should be noted.

- The amount of storage required is proportional to the number of states, and to the length of the trellis in symbols (N).
- The computational complexity is dominated by the number of states.
- Because the computation of the $B_t(k)$ terms involves the magnitude of the matched filter output, one may be able to reduce the number of filters for a given modulation scheme by identifying filters which will produce the same output.

For a given composite fading plus noise power spectral density, the matched filters are FIR filters with fixed coefficients. Under changing channel conditions, these filters will have to be updated periodically by some sort of adaptive channel estimator. The sensitivity of the algorithm to errors in the model was investigated in [20]; it was found to be largely insensitive to errors in the modeled additive noise level, and moderately sensitive to errors in the modeled fading bandwidth.

III. Demodulation Performance

A. Description of Simulation

The performance of the symbol-by-symbol MAP demodulator was evaluated using Monte Carlo simulation. Three different receivers were implemented, allowing the performance of the MAP receiver to be compared to that of the linear predictive (MLSE) receiver [12] and that of conventional differential detection.

B. Demodulation of Binary CPM Signals

For binary CPM, the bandwidth-efficient N32FM pulse shape described in [21] was used. This pulse shape satisfies Nyquist's second and third criteria for no intersymbol interference and, because CPM is inherently differentially encoded, it can be demodulated effectively via differential detection. It was assumed that the receiver downconverts the signal to complex baseband, passes it through an ideal antialiasing filter, and then samples it at $r = 2$ samples/symbol.

Simulations were performed to compare the performance of the MAP demodulator with that of the MLSE-based linear predictive receiver and that of differential detection. Two fading channel rates were investigated, $0.05R$ and $0.3R$, where R is the symbol rate. The general simulation parameters were

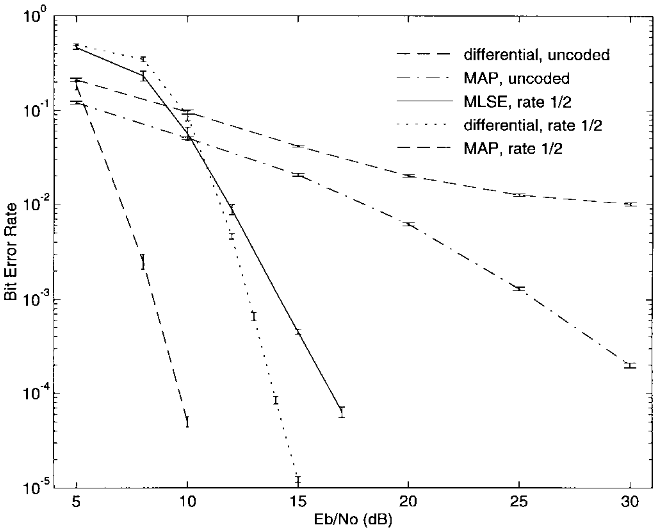


Fig. 4. BER performance of the MAP and MLSE demodulators as compared to that of differential detection for the reception of binary CPM signals. The fading rate is $0.05R$. Coded MLSE results utilize hard decision inputs to the decoder, while the coded MAP and differential detection results utilize soft decisions.

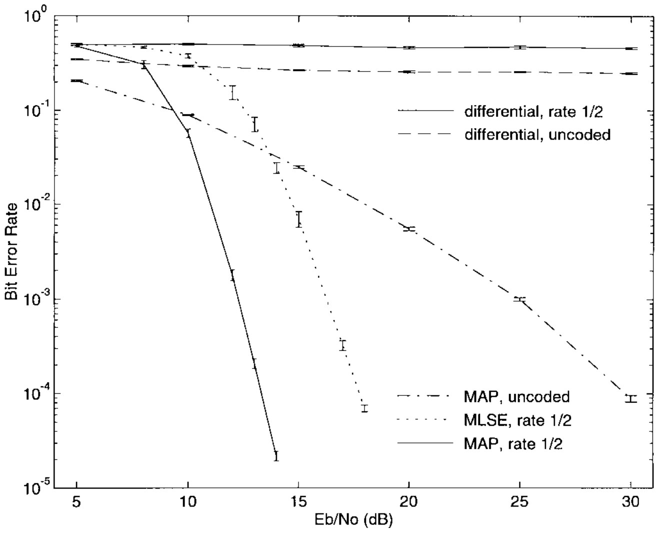


Fig. 5. BER performance of the MAP and MLSE demodulators as compared to that of differential detection for the reception of binary CPM signals. The fading rate is $0.3R$. Coded MLSE results utilize hard decision inputs to the decoder, while the coded MAP and differential detection results utilize soft decisions.

as follows:

fading rate:	$0.05R$, $0.3R$
sample rate:	2 samples/sysmbol
CPM pulse shape:	N32FM ($\alpha = 2.5$ symbol periods)
predictor:	fifth order ($\beta = 2.5$ symbol periods)
block length (N):	1024 bits.

The number of MAP states was $2^{\lceil 2.5+2.5 \rceil -1} = 16$. The resulting BER performance is shown in Figs. 4 and 5, respectively. A number of points can be noted.

- Differential detection results in an irreducible error floor that cannot be improved upon by increasing the signal-to-noise ratio.
- Differential detection is essentially useless at a fading rate of $0.3R$.
- There is no sign of an error floor in the MAP results, even at an E_b/N_0 of 30 dB.

Although not shown in Figs. 4 and 5, the uncoded performance of the MLSE demodulator also was simulated. The difference between the uncoded performance of the MAP and MLSE processing is negligible, which is expected given the known performance of the Viterbi algorithm.

Also not shown in Figs. 4 and 5 are the BER estimates produced by the MAP demodulator. Because the symbol-by-symbol MAP algorithm inherently produces reliability estimates on each symbol, it is a simple matter to generate a raw BER estimate by transforming the symbol reliability estimates into bit reliability estimates. In the simulations of Figs. 4 and 5, the BER estimates produced by the MAP algorithm closely matched the BER's measured in the simulation, indicating that the assumptions upon which the MAP processing is based are generally valid. The BER estimation capability may be useful for applications requiring channel performance monitoring or power control.

In order to evaluate the effectiveness of the soft decisions produced by the symbol-by-symbol MAP demodulator, the output probabilities from the demodulator were fed to a convolutional decoder. The code utilized was the industry standard rate-1/2, constraint length-7 convolutional code [22]. A 128 row × 8 column interleaver was used to decorrelate the bit errors prior to decoding. While this interleaver does not represent ideal interleaving, it does provide most of the associated gain while remaining a practical length.

The performance of MAP demodulation is compared with that of soft decision decoding of differentially detected CPM in Figs. 4 and 5. As a reference, the performance of hard decision decoding using the outputs of the linear predictive receiver is also shown. From Fig. 4, a number of points are apparent.

- The simple soft decisions from a differential detector work quite well at a fading rate of $0.05R$.
- The improved soft decisions from the MAP demodulator provide over 4 dB of additional gain at an error rate of 10^{-5}.
- The soft decisions from the MAP demodulator provide over 7 dB of gain relative to the hard decisions from the MLSE demodulator.

When the results shown in Fig. 5 for the fast fading channel are examined, we can see the following.

- The soft decisions from the differential detector no longer are capable of driving the Viterbi decoder.
- The soft decisions from the MAP demodulator continue to provide good performance, exhibiting over 5 dB of gain relative to the hard decisions from the MLSE demodulator at an error rate of 10^{-5}.

C. Demodulation of QPSK Signals

In this section, we consider the performance of symbol-by-symbol MAP demodulation of QPSK signals. At 1 sample/symbol, QPSK can be considered to be a discrete-time form of CPM.

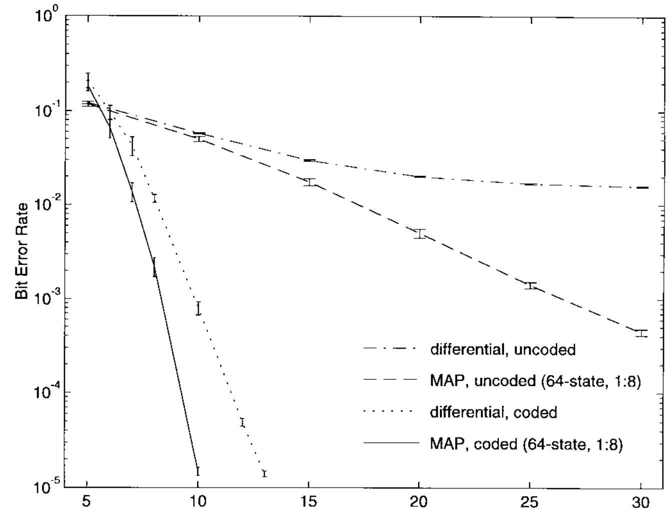


Fig. 6. BER performance of the MAP demodulator as compared to that of differential detection for the reception of QPSK signals. The fading rate is $0.05R$.

An important consideration in dealing with QPSK signals (in the absence of a phase reference, or differential encoding) is how to deal with the fourfold phase ambiguity. One approach, chosen here, is to periodically insert known pilot symbols into the transmitted symbol stream. Note that, unlike the case of pilot-symbol-assisted modulation [10], the pilot symbol rate is not a function of the fading rate. This will reduce the bandwidth penalty which is incurred by the pilot symbols on fast fading channels.

A number of simulations were performed to evaluate the optimum pilot symbol spacing. As a compromise between performance and bandwidth overhead, a pilot symbol rate of 1 in 9 (1 : 8) was chosen for further study.

Simulations were performed to compare the performance of the MAP demodulator with that of differential detection of differentially encoded QPSK. The simulation parameters were as follows:

fading rate:	$0.05R$
sample rate:	1 sample/sysmbol
symbol encoding:	Gray coding
QPSK pulse shaping:	none ($\alpha = 1$ symbol period)
MAP predictor order:	third order ($\beta = 3$ symbol periods)
MAP pilot symbol rate:	1:8
block length (N):	4096 bits.

The number of MAP states was $4^{\lceil 1+3 \rceil - 1} = 64$. The resulting BER performance is shown in Fig. 6.

A number of points can be noted.

- As with the binary CPM case, differential detection results in an irreducible error floor that cannot be improved upon by increasing the signal-to-noise ratio.
- There is no sign of an error floor in the MAP results, even at an E_b/N_0 of 30 dB.

Not shown in Fig. 6 are the BER estimates produced by the MAP demodulator. As with the binary CPM case, the BER estimates computed from the reliability estimates produced by the MAP algorithm closely matched the BER's measured in the simulation.

The error bursts produced by the MAP demodulator are longer in duration than those obtained using differential detection. This suggests that the MAP error bursts are not just caused by the channel entering a deep fade, but are instead caused by phase slips. The resulting error runs dominate the bit error rate. Note that the length of an error run due to a phase slip can exceed the pilot symbol spacing since the pilot symbol only reduces the number of possible states from 64 to 16. Three contiguous pilot symbols would be required to guarantee resolution of the phase ambiguity every pilot symbol period.

Fig. 6 also illustrates the performance of the symbol-by-symbol MAP demodulator when utilized with a following convolutional decoder. Once again, the code utilized was the industry standard rate-1/2, constraint length-7 convolutional code [22]. A 128 row × 32 column interleaver was used to decorrelate the bit errors prior to decoding (the interleaver size was chosen to minimize loss in the iterative configuration of Section IV-C). The results show that the improved soft decisions from the MAP demodulator provide approximately 3 dB of gain at an error rate of 10^{-5}.

While rectangular pulse-shaped QPSK was used for the above simulations, the results generally are applicable to any scenario where the fading does not decorrelate the pulse after matched filtering at the receiver, i.e., where a sample rate of 1 sample/symbol is valid. From [23], a fading rate of $0.05R$ requires roll-offs of approximately 50% or greater for this assumption to be valid. This requirement is consistent with the modulations employed in many mobile communications systems.

IV. Iterative MAP Processing for Joint Demodulation and Decoding

In this section, we examine the possibility of using iterative demodulation and decoding to improve the performance of the symbol-by-symbol MAP demodulator. Because it utilizes *a priori* probabilities, and because the soft decisions produced by the demodulator are inherently symbol (or bit) probabilities, the MAP demodulator is well suited to iterative processing applications where the demodulator is explicitly included in the iterative processing.

A. Motivation

Increasingly, communications systems are being composed of a long cascade of signal processing intensive subsystems. Fig. 7 illustrates an example that is typical of the type of cascaded arrangement being introduced for many radio applications. In general, the subsystems in the receiver perform inverse operations to corresponding subsystems in the transmitter. In a "conventional" receiver, each subsystem passes a sequence of bits to the following subsystem. The problem with this approach is that by outputting bits (i.e., hard decisions), information is lost in each subsystem. This is because, while the subsystem only indicates whether it believes that a given bit is a 0 or a 1, it usually has sufficient information

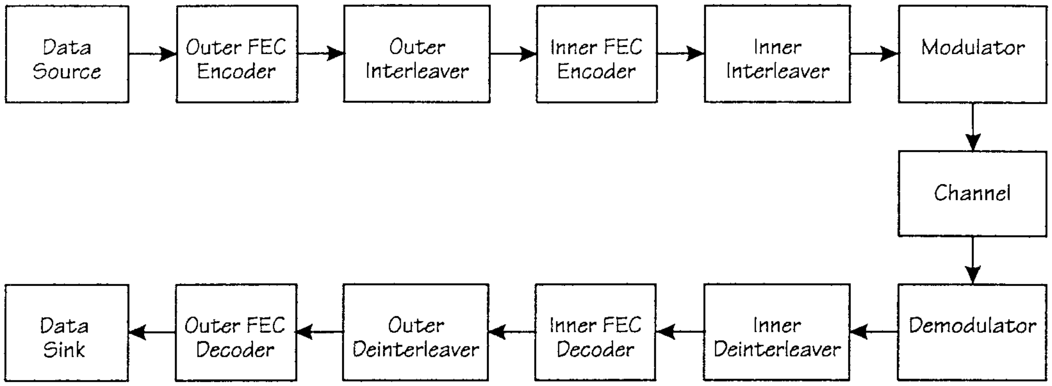


Fig. 7. Example of a system utilizing concatenated processing techniques.

to estimate the degree of confidence in its decisions. One straightforward way to reduce the loss of information, and the resulting loss in performance, is to pass the confidence level along with the decision (i.e., soft decisions). This is often done when passing information from a demodulator to an inner decoder, which is known to result in approximately a 2-dB performance benefit in the additive white Gaussian noise (AWGN) channel [24]. However, even if optimal bit-by-bit soft decisions are passed between all of the subsystems in the receiver, the overall performance can be far from optimal. This results from the fact that, while later stages can use the information gleaned from previous stages, the reverse is not generally true. While one means of achieving optimal performance is to perform a joint detection, taking all receive processing into account simultaneously, the complexity of such an overall approach is usually prohibitive. This motivates an iterative processing approach which allows earlier stages to refine their processing based on information from later stages.

B. General Concept

The operation of an iterative processing system can be thought of in terms of the information that each stage adds to the final result. As an example, consider the operation of a demodulator. The goal of the demodulator is to produce soft decisions which reflect the probability that a given bit is a 0 or a 1. The information available to the demodulator is the received signal $y(t)$ and the initial *a priori* probabilities of the input bits (which are generally 1/2). The demodulator uses this information, combined with knowledge of the chosen modulation and of the channel structure, to produce its soft decisions.

Viewed mathematically on a bit-by-bit basis, the demodulator wishes to determine the *a posteriori* bit probabilities

$$\Pr(u_n = 0|y(t)) = \frac{p(y(t)|u_n = 0) \cdot \Pr(u_n = 0)}{p(y(t))} \tag{24}$$

and

$$\Pr(u_n = 1|y(t)) = \frac{p(y(t)|u_n = 1) \cdot \Pr(u_n = 1)}{p(y(t))} \tag{25}$$

for all $\{u_n\}$ (this is essentially the same as (2) in the derivation of the symbol-by-symbol MAP demodulator). Equation (25) embodies the bit soft decision as represented thus far in this paper. Two other equally valid representations are the likelihood ratio and the log-likelihood ratio. Consider the likelihood ratio formed from the *a posteriori* probabilities of (24) and (25):

$$\begin{aligned} L_n &= \frac{\Pr(u_n = 1|y(t))}{\Pr(u_n = 0|y(t))} \\ &= \frac{p(y(t)|u_n = 1)}{p(y(t)|u_n = 0)} \bullet \frac{\Pr(u_n = 1)}{\Pr(u_n = 0)}. \end{aligned} \tag{26}$$

We can see that this ratio is, in fact, the product of two distinct quantities. The first term, being conditioned on a particular value of u_n, is independent of the *a priori* probabilities for the bit at time n, although it may be dependent upon the *a priori* probabilities for bits transmitted at times other than time n. The second term embodies the *a priori* probabilities for u_n.

For clarity of exposition, it is useful to rewrite (26) using logarithms of the likelihood ratios. Converting to a vector notation to eliminate the n subscript, we get

$$\boldsymbol{R}^c = \boldsymbol{R}^i + \boldsymbol{R}^a \tag{27}$$

where $\boldsymbol{R}^c$ is the composite log-likelihood ratio, $\boldsymbol{R}^i$ is the log-likelihood ratio of the so-called *intrinsic* information (a term introduced in [18] to refer to the information that the demodulator gleans about u_n from the input signal and potentially the *a priori* probabilities of the other transmitted bits, without utilizing the *a priori* probability for u_n), and $\boldsymbol{R}^a$ is the log-likelihood ratio of the *a priori* information.

As an example, consider the receiver portion of Fig. 7 where the demodulator is stage 0, the first decoder is stage 1, and the second decoder is stage 2. The inputs to the first decoder are the soft decisions embodying the *a priori* bit probabilities and the *intrinsic* information obtained by the demodulator. The decoder utilizes its knowledge of the code structure to produce a refined set of soft decisions (input bit probabilities). From an information perspective, the output of this stage can be represented as

$$\boldsymbol{R}^c(1) = \boldsymbol{R}^i + \boldsymbol{R}^a + \boldsymbol{R}^e(1) \tag{28}$$

where $\boldsymbol{R}^c(1)$ is the composite log-likelihood ratio after the processing of stage 1 (the first decoder), and $\boldsymbol{R}^e(1)$ is the *extrinsic* information added by stage 1 (information added by a stage other than the demodulator which is independent of the input soft decision at time n, but utilizes the soft decisions for other bits). When $\boldsymbol{R}^c(1)$ is passed as the input to stage 2 (the second decoder), the sum $\boldsymbol{R}^a + \boldsymbol{R}^e(1)$ can be viewed as augmented *a priori* information to be utilized by stage 2.

It is important to note that (28) only holds if the inputs to the decoder are independent. If the inputs are not independent, then the log-likelihood ratios may not be factored into separate terms. If the channel has memory, this independence assumption will not be valid; therefore, interleaving must be present between the demodulator and the decoder. The same argument holds for subsequent decoder stages. Since the *extrinsic* information produced by a decoder for the bit at time n utilized the data bit *a priori* probabilities and the *intrinsic* information from the demodulator for all times $k \neq n$, the *extrinsic* information for the bits at any two times n_1 and n_2 may be correlated. Fortunately, this correlation decreases as $|n_1 - n_2|$ increases; therefore, interleaving can be utilized to spread correlated information outside the memory of the subsequent decoder stage. In essence, a diversity effect is achieved since the *extrinsic* information at time n is only weakly correlated to the *a priori* bit probability and the *intrinsic* information at time n.

The above concepts can be extended for any D-stage system utilizing soft-in soft-out processing, i.e., the composite log-likelihood ratio out of stage d can be written

$$\boldsymbol{R}^c(d) = \boldsymbol{R}^i + \boldsymbol{R}^a + \sum_{j=1}^{d} \boldsymbol{R}^e(j) \qquad (29)$$

for $1 \leq d \leq D$.

Consider now an iterative processing approach which includes the demodulator. *In iterative processing, the information input to a stage on the second and subsequent iterations must not include the information added by that stage in the previous iteration*. In the case of the demodulator, the *a priori* information utilized therefore includes the initial *a priori* information and the *extrinsic* information added by the subsequent stages on the previous iteration, but does not include the *intrinsic* information previously determined by the demodulator. One way that this can be achieved is for the demodulator to store its *intrinsic* information vector from the most recent iteration and to subtract it from the composite information vector, received from the Dth stage, prior to computing the new *intrinsic* information vector. For the pth iteration, the augmented *a priori* information input to the demodulator therefore can be written

$$\boldsymbol{R}_p^a(0) = \boldsymbol{R}^a + \sum_{j=1}^{D} \boldsymbol{R}_{p-1}^e(j) \qquad (30)$$

while the corresponding composite log-likelihood ratio out of the demodulator becomes

$$\boldsymbol{R}_p^c(0) = \boldsymbol{R}_p^i + \boldsymbol{R}^a + \sum_{j=1}^{D} \boldsymbol{R}_{p-1}^e(j) \qquad (31)$$

where $\boldsymbol{R}_p^i$ is the augmented *intrinsic* information provided by the demodulator on the pth iteration.

The inputs to stage d on the pth iteration are $\boldsymbol{R}_p^i$ and the augmented *a priori* information is given by

$$\boldsymbol{R}_p^a(d) = \boldsymbol{R}^a + \sum_{j=1}^{d-1} \boldsymbol{R}_p^e(j) + \sum_{j=d+1}^{D} \boldsymbol{R}_{p-1}^e(j). \qquad (32)$$

Note that the augmented *a priori* information includes the initial *a priori* information and the most recent *extrinsic* information available from all other stages, but does not include any *extrinsic* information from stage d. One way that this can be achieved is for stage d to store its *extrinsic* information vector from the most recent iteration and to subtract it from the composite information vector, received from the $(d-1)$th stage, prior to computing the new *extrinsic* information vector.

The corresponding composite log-likelihood ratio out of stage d is then

$$\boldsymbol{R}_p^c(d) = \boldsymbol{R}_p^i + \boldsymbol{R}^a + \sum_{j=1}^{d} \boldsymbol{R}_p^e(j) + \sum_{j=d+1}^{D} \boldsymbol{R}_{p-1}^e(j). \qquad (33)$$

C. Performance with QPSK Signals

Simulations were performed to determine the performance of the iterative MAP receiver with QPSK signals. The simulation parameters were the same as for the noniterative simulations. The convolutional code is processed using a MAP filter for all iterations except the last, where a MAP decoder is used. A MAP filter refers to the application of MAP processing to determine the *a posteriori* probability of the *coded* bits rather than of the *information* bits [16], [17].

As in Section III-C, the pilot symbols are simply treated as symbols whose *a priori* probability is known to be one. The iterative processing makes use of the *extrinsic* information generated by the MAP filter from its knowledge of the convolutional code structure to improve the augmented *a priori* probabilities for nonpilot symbols fed to the demodulator on subsequent passes. The improved soft decisions output by the demodulator help to refine the *extrinsic* information generated by the MAP filter, and so on, until the MAP decoder is used to output the final information bits.

The resulting BER performance is shown in Fig. 8. Note that “1 iteration” refers to the normal noniterative approach, i.e., the results illustrated in Section III. Also shown in the figure is the performance of a fictitious reference receiver utilizing ideal channel state information. This receiver produces perfect soft decisions, i.e., soft decisions which have no phase ambiguity and whose magnitude is proportional to the square of the amplitude change introduced by the channel. No differential encoding or pilot symbols are required.

A number of points can be noted.

- Most of the improvement (1–1.5 dB in the range of interest) is obtained with a single additional iteration.
- While iterating provides some performance improvement at low E_b/N_0, the additive noise clearly inhibits the ability of the iterative processing to converge to accurate channel estimates (this can be seen by noting the significantly better performance of the ideal channel state receiver at these E_b/N_0).

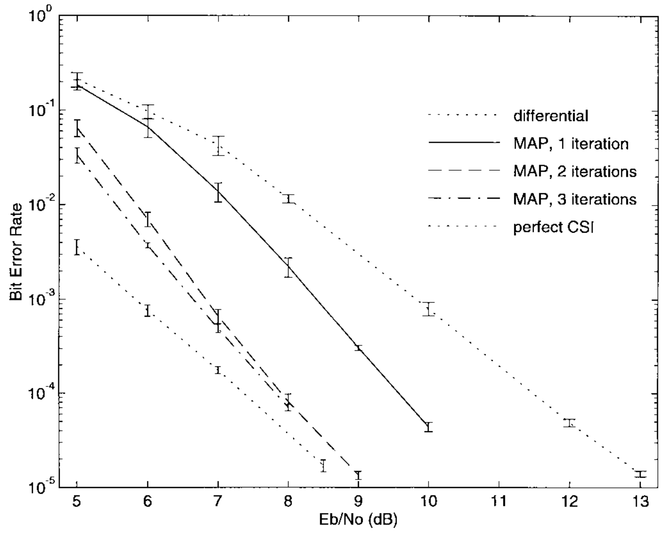


Fig. 8. BER performance of iterative MAP demodulation/decoding as compared to that of conventional differential detection and convolutional decoding for the reception of QPSK signals. Also shown is the performance of a fictitious reference receiver utilizing ideal channel state information. The fading rate is $0.05R$, and the pilot symbol rate used in the MAP simulations is 1 : 8.

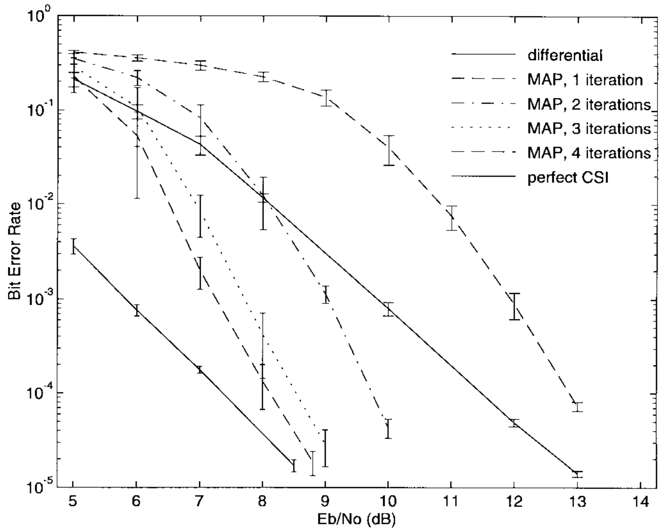


Fig. 9. BER performance of iterative MAP demodulation/decoding as compared to that of conventional differential detection and convolutional decoding for the reception of QPSK signals. Also shown is the performance of a fictitious reference receiver utilizing ideal channel state information. The fading rate is $0.05R$, and the pilot symbol rate used in the MAP simulations is 1 : 16.

- At an error rate of 10^{-5}, the performance of the MAP receiver utilizing two iterations approaches that of decoding with ideal channel state information.

This last result is particularly impressive. When the 0.5 dB overhead due to power in the pilot symbols is taken into account, the performance of the iterative MAP receiver matches, within error bars, that of the receiver utilizing ideal channel state information.

Since the performance delta between the iterative MAP receiver and the reference receiver is essentially wholly due to the power in the pilot symbols, it is desirable to reduce the pilot symbol rate, and therefore the corresponding power penalty. To this end, a system utilizing a 1 : 16 pilot symbol rate was investigated. The resulting BER performance is shown in Fig. 9. A number of points can be noted.

- The single iteration performance is worse than that obtained using simple differential detection.
- Most of the improvement (4–5 dB in the range of interest) is obtained with two or three additional iterations.
- At an error rate of 10^{-5}, the performance of the MAP receiver utilizing four iterations approaches that of decoding with ideal channel state information.

Once again, this last result is particularly impressive. When the 0.26 dB of power in the pilot symbols is taken into account, the performance of the iterative MAP receiver matches, within error bars, that of the receiver utilizing ideal channel state information.

The difference in the results for the 1 : 8 and 1 : 16 pilot symbol rates shows that the iterative processing is particularly effective in combating the effect of phase slips; this was a result which was anticipated. To see why, one can look at the performance of a convolutional decoder in a noniterative system. When a decoder is operating above its threshold E_b/N_0, its decisions tend to have occasional error bursts interspersed with largely correct data; this property is amplified as the E_b/N_0 increases. Since a MAP filter is essentially a decoder without its final decoding stage, its output exhibits the same characteristic, except that the error bursts are bursts of bits whose confidence level is low (i.e., bits with $\Pr(x = 1) \approx 0.5$). When the output of the MAP filter is reinterleaved, both the low-confidence and high-confidence probabilities are distributed across the interleaver block. Bits with probabilities near 0.5 generally will be associated with bits which had similar probabilities out of the demodulator or occurred during phase slips (and therefore failed to line up with the FEC code). More importantly, the majority of the bits (which should have probabilities near one) serve as almost-known symbols, and aid in resolving phase ambiguities in the same way that pilot symbols do.

V. Summary

The symbol-by-symbol MAP demodulator exhibits good performance over the range of channel conditions studied. In an uncoded system, no irreducible error rate is observed, even for an E_b/N_0 of 30 dB at a fading rate of $0.3R$, and the BER estimates produced by the demodulator closely match the measured BER. While pilot symbols are required for use with QPSK, their rate is determined only by phase ambiguity considerations, and not the fading rate.

The performance results for the symbol-by-symbol MAP demodulator feeding a convolutional decoder indicate that the demodulator's soft decisions produce a 3–4 dB performance improvement over differential detection at the lower fading rate ($0.05R$,) and remain robust at the higher fading rate ($0.3R$) when the differential detector is unusable. For a 1 : 8 pilot symbol rate, a further 1–1.5 dB of improvement was observed when a single additional iteration of the demodulation/decoding process was performed on a QPSK signal, while up to 5 dB of improvement was obtained with three additional iterations at a 1 : 16 pilot symbol rate. In both cases, the resulting performance is within a small fraction of a decibel of that of a receiver utilizing ideal channel state information! No other published technique is capable of providing performance so close to ideal in fast-fading environments.

While significantly more computationally intensive than an MLSE approach, iterative processing of multiple MAP stages can approximate jointly optimal decoding closely without the corresponding increase in complexity required by the latter. Low-complexity techniques for soft-in/soft-out processing is an active and important field of research, and techniques such as the soft-output Viterbi algorithm [27] can be used to decrease the complexity with only small impacts on performance.

The techniques reported in this paper follow from those of [12], which assumes constant modulus signaling and flat fading. It is anticipated that these techniques can be generalized to nonconstant modulus signaling and frequency-selective fading in ways similar to the generalizations of [12] published in [25] and [26].

References

[1] E. Reinhart, R. Taylor, A. Heyward, and J. Miller, "WARC's last act?," *IEEE Spectrum*, vol. 29, pp. 20–33, Feb. 1992.

[2] A. Viterbi, "Wireless digital communication: A view based on three lessons learned," *IEEE Commun. Mag.*, vol. 29, pp. 33–36, Sept. 1991.

[3] J. Anderson, T. Aulin, and C.-E. Sundberg, *Digital Phase Modulation.* New York: Plenum, 1986.

[4] T. Aulin and C.-E. Sundberg, "Continuous phase modulation, Part I: Full response signaling," *IEEE Trans. Commun.*, vol. COM-29, pp. 196–209, Mar. 1981.

[5] ——, "Continuous phase modulation, Part II: Partial response signaling," *IEEE Trans. Commun.*, vol. COM-29, pp. 210–225, Mar. 1981.

[6] J. Anderson and C.-E. Sundberg, "Advances in constant envelope coded modulation," *IEEE Commun. Mag.*, vol. 29, pp. 36–45, Dec. 1991.

[7] W. Stutzman, "Prolog to Special Section on Propagation Effects on Satellite Communication Links," *Proc. IEEE*, vol. 81, pp. 850–855, June 1993.

[8] J. G. Proakis, *Digital Communications.* New York: McGraw-Hill, 1983.

[9] W. Y. C. Lee, *Mobile Communications Engineering.* New York: McGraw-Hill, 1982.

[10] J. Lodge and M. Moher, "Time diversity for mobile satellite channels using trellis coded modulation," in *Proc. IEEE GLOBECOM'87*, pp. 303–307.

[11] J. Cavers, "An analysis of pilot symbol assisted modulation for Rayleigh fading channels," *IEEE Trans. Veh. Technol.*, vol. 40, pp. 686–693, Nov. 1991.

[12] J. Lodge and M. Moher, "Maximum likelihood sequence estimation of CPM signals transmitted over Rayleigh flat-fading channels," *IEEE Trans. Commun.*, vol. 38, pp. 787–794, June 1990.

[13] J. Hagenauer, "Soft-in/soft-out—The benefits of using soft values in all stages of digital receivers," in *3rd Int. Workshop Digital Signal Processing Techniques Applied Space Commun.*, 1992.

[14] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. Inform. Theory*, vol. IT-20, pp. 284–287, Mar. 1974.

[15] G. D. Forney, Jr., "The Viterbi algorithm," *Proc. IEEE*, vol. 61, pp. 268–278, Mar. 1973.

[16] J. Lodge, P. Hoeher, and J. Hagenauer, "The decoding of multidimensional codes using separable MAP filtering," in *Proc. 16th Biennial Symp. Commun.*, 1992, pp. 343–346.

[17] J. Lodge, R. Young, P. Hoeher, and J. Hagenauer, "Separable MAP 'filters' for the decoding of product and concatenated codes," in *Proc. IEEE ICC'93* pp. 1740–1745.

[18] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error correcting coding and decoding: Turbo codes(1)," in *Proc. IEEE ICC'93*, pp. 1064–1070.

[19] A. Papoulis, *Probability, Random Variables, and Stochastic Processes.* New York: McGraw-Hill, 1984.

[20] M. Gertsman, "Symbol-by-symbol MAP demodulation of CPM signals on Rayleigh flat fading channels," M. Eng. thesis, Dep. Syst. Comput. Eng., Carleton Univ., Ottawa, Canada, 1996.

[21] B. Sayer and S. Pasupathy, "Nyquist 3 pulse shaping in continuous phase modulation," *IEEE Trans. Commun.*, vol. COM-35, pp. 57–67, Jan. 1987.

[22] Qualcomm Inc., "Q0256 $k = 7$ multi-code rate Viterbi decoder," Tech. Data Sheet, June 1990.

[23] J. Cavers, "On the validity of the slow and moderate fading models for matched filter detection of Rayleigh fading signals," *Can. J. Elect. Comput. Eng.*, vol. 17, no. 4, 1992.

[24] A. J. Viterbi and J. K. Omura, *Principles of Digital Communications and Coding.* New York: McGraw-Hill, 1979.

[25] Q. Dai and E. Shwedyk, "Detection of bandlimited signals over frequency selective Rayleigh fading channels," *IEEE Trans. Commun.*, vol. 42, pp. 941–950, Feb./Mar./Apr. 1994.

[26] X. Yu and S. Pasupathy, "Innovations-based MLSE for Rayleigh fading channels," *IEEE Trans. Commun.*, vol. 43, pp. 1534–1544, Feb./Mar./Apr. 1995.

[27] J. Hagenauer and P. Hoeher, "A Viterbi algorithm with soft-decision outputs and its applications," in *Proc. IEEE GLOBECOM'89*, pp. 47.11–47.17.

Michael J. Gertsman (M'90) received the B.A.Sc. degree in electrical engineering from the University of British Columbia, Vancouver, BC, in 1987 and the M.Eng. degree in systems and computer engineering from Carleton University, Ottawa, Ont., Canada, in 1997.

In 1987, he joined the Public Switching division of Mitel Corporation, Kanata, Ont., where he focused on signaling protocols. While at Mitel, he was awarded a patent for his work on R2 interregister signaling. In 1990, he joined SkyWave Electronics Ltd., Kanata, Ont. (acquired in 1993 by Calian Communications Systems Ltd.) where he designed and implemented DSP-based satellite communications equipment for the aeronautical, maritime, and land mobile environments. In 1995, he participated in the launch of a new company, Square Peg Communications Inc., Ottawa, Ontario. As Director, Mobile Satellite Products at Square Peg, his responsibilities include systems definition, engineering development, and marketing of advanced mobile satellite communication products. His research interests lie in the application of digital signal processing techniques to radio modems.

John H. Lodge (S'80–M'81–SM'90) received the B.Sc. and Ph.D. degrees in electrical engineering from Queen's University, Kingston, Ont., Canada, in 1977 and 1981, respectively.

From September 1981 to April 1984, he was with the Advanced Systems Division of Miller Communications Systems Ltd., Kanata, Ont., Canada, where he was involved in the analysis and design of HF, spread-spectrum, and satellite communications systems. Since May 1984, he has been with the Mobile and Personal Communications Division at the Communications Research Centre (CRC). He is currently the Project Leader of Communications Signal Processing for that division. His group has performed design and analysis for mobile satellite communication systems for numerous private sector clients. They are also active in supporting Canadian government technical requirements such as spectrum monitoring and military communication. He is an Adjunct Professor of the University of Ottawa and the University of Manitoba. He serves as the CRC member on technical review committees on mobile and satellite communications for the Telecommunications Research Institute of Ontario, and for the Canadian Institute for Telecommunications Research.