Optimal Sequence Detection and Optimal Symbol-by-Symbol Detection: Similar Algorithms

JEREMIAH F. HAYES, SENIOR MEMBER, IEEE, THOMAS M. COVER, FELLOW, IEEE, AND JUAN B. RIERA, MEMBER, IEEE

Abstract—**An algorithm is derived which performs optimal symbol-by-symbol detection of a pulse amplitude modulated sequence. The algorithm is similar to the Viterbi algorithm with the optimality criterion optimal symbol detection rather than optimal sequence detection. A salient common feature is the merge phenomenon which allows common decisions to be made before the entire sequence is received.**

INTRODUCTION

THE Viterbi algorithm [1] is a dynamic programming technique for decoding sequential codes. Forney [2], [3] applied the algorithm to the detection of pulse amplitude modulated sequences disturbed by additive Gaussian noise and intersymbol interference. Optimal detection of an entire transmitted sequence is achieved while the number of computations grows linearly with the length of the sequence. In the sequel we shall derive an algorithm which uses the entire received sequence for symbol-by-symbol detection with minimum probability of error. The algorithm was inspired by a version of the Viterbi algorithm and, as a consequence, has a number of similar features. In particular, for both algorithms there is a merge phenomenon which allows optimum detection before the entire sequence is received. Earlier work by Abend and Fritchman [4] considered optimal symbol detection using a portion of the received sequence. Related work was also done by Chang and Hancock [5], who developed a sequential procedure for making optimum decisions about a subsequence of symbols whose length is equal to the memory in the channel.

The Viterbi algorithm offers considerable advantage over nonsequential detection algorithms, which grow in computational complexity exponentially with the length of the sequence. Nevertheless the algorithm, in its pure form, is still complex. If the channel memory is m, and L symbol levels are transmitted, the receiver must store and process L^m state variables. Considerable effort has been expended on proposing and evaluating approximations to the Viterbi algorithm which reduce this complexity. One approach [6], [7] has been to prefilter the received waveform so as to reduce the channel memory. A second approach has been to eliminate from consideration a subset of the internal states [8], [9]. A series of papers [10]-[13] has delved further into the problems of practical implementation, treating aspects such as unknown channel characteristics and microprocessor implementation. The Viterbi algorithm is shown to be robust [10] in the sense that approximations do not lead to a precipitous deterioration in performance.

As we shall see, the symbol-by-symbol algorithm is rather complex. However, the approaches that have been taken to approximate the Viterbi algorithm can also be employed here. Hopefully, similar strides toward practical application can be taken.

MATHEMATICAL BACKGROUND

We consider the case of data transmitted at baseband[1] by means of pulse amplitude modulation [14]. The transmitted sequence is $a_j; j = 1, 2, \cdots, N$. We assume that the transmitted symbols a_j are independent of one another and may assume L different values with equal probability. The length N of the transmitted sequence is assumed to be large but finite. The symbols are transmitted at a rate of $1/T$ symbols/s over a linear channel perturbed by additive white Gaussian noise.[2] The received signal is

$$y(t) = \sum_{j=1}^{N} a_j h(t - jT) + n(t); \quad 0 \leqslant t \leqslant \tau, \ nT \leqslant \tau < \infty \quad (1)$$

where the impulse response $h(t)$ represents the combined effects of the channel and shaping filters at the transmitter and receiver. The Gaussian noise with power density spectrum $N_0/2$ is denoted by $n(t)$. We assume that $h(t)$ has finite duration, allowing us to confine the received signal to the interval $[0, \tau]$.

In the succeeding sections we shall examine, in turn, optimum sequence detection and optimum bit-by-bit detection. The development of optimum sequence detection follows that of Ungerboeck [15] and Mackechnie [16]. While this derivation has been presented elsewhere [20], we shall give an ab-

Paper approved by the Editor for Communication Theory of the IEEE Communications Society for publication without oral presentation. Manuscript received May 15, 1975; revised June 2, 1981. This work was supported in part by the National Sciences and Engineering Research Council of Canada under Grant A0901, the National Science Foundation under Grant ECS78-2334, and the Joint Services Electronics Program under Grant DAAG-29-79-C-0047.

J. F. Hayes was with Bell Laboratories, Murray Hill, NJ 07974. He is now with the Department of Electrical Engineering, McGill University, Montreal, P.Q., Canada H3A 2A7.

T. M. Cover was with Bell Laboratories, Murray Hill, NJ 07974. He is now with the Departments of Electrical Engineering and Statistics, Stanford University, Stanford, CA 94305.

J. B. Riera is with the Telecommunications School and Faculty of Informatics, Universidad Politecnica de Madrid, Madrid, Spain.

[1] We shall address ourselves exclusively to the problem of baseband signal detection. Ungerboeck [17] has extended the Viterbi algorithm to the passband channel. This technique can be used to extend our results in a similar fashion.

[2] The white noise assumption simplifies the analysis considerably. However, similar results can be obtained for nonwhite noise.

0090-6778/82/0100-0152$00.75

breviated version for the sake of completeness. Both algorithms have certain common elements which will now be presented. A key term that arises in the decision rules is the probability density functional

$$\Pr\,[y(t);\ 0 \leqslant t \leqslant \tau \,|\, a_1 = \hat{a}_1, a_2 = \hat{a}_2, \cdots, a_N = \hat{a}_N].$$

This is the probability of a particular signal being received in the interval $[0, \tau]$ given that a particular sequence of symbols $\hat{a}_1, \hat{a}_2, \cdots, \hat{a}_N$ was transmitted. Since the impulse response is known, the only random component of the received signal is the additive channel noise [see (1)] which is assumed to be white and Gaussian. Under these conditions the expression for this conditional probability is

$$\Pr\,[y(t),\ 0 \leqslant t \leqslant \tau \,|\, a_1 = \hat{a}_1, \hat{a}_2 = a_2, \cdots, a_N = \hat{a}_N]$$

$$= K \exp\left[-\frac{1}{2N}\int_0^\tau \left[y(t) - \sum_{i=N}^{N} \hat{a}_i h(t - iT)\right]^2 dt\right]. \quad (2)$$

Our interest in (2) lies in the relative values for different sequences $\hat{a}_1, \hat{a}_2, \cdots, \hat{a}_N$. Therefore, we concentrate on the exponent. This exponent can be decomposed into the convenient form

$$\int_0^\tau \left[y(t) - \sum_{i=1}^{N} \hat{a}_i h(t - iT)\right]^2 dt$$

$$= \int_0^\tau y^2(t)\,dt - 2\sum_{i=1}^{N} \hat{a}_i z_i + \sum_{i=1}^{N}\sum_{j=1}^{N} \hat{a}_i \hat{a}_j r_{i-j}$$

$$= \int_0^\tau y^2(t)\,dt - 2\sum_{i=1}^{N-1} \hat{a}_i z_i + \sum_{i=1}^{N-1}\sum_{j=1}^{N-1} \hat{a}_i \hat{a}_j r_{1-j}$$

$$- 2\hat{a}_N z_N + 2\hat{a}_N \sum_{i=1}^{N-1} \hat{a}_i r_{N-j} + \hat{a}_N{}^2 r_0 \quad (3)$$

where

$$z_i \triangleq \int_0^\tau z(t) h(t - iT)\,dt$$

and

$$r_{i-j} \triangleq \int_0^\tau h(t - iT) h(t - jT)\,dt.$$

We make the assumption that the intersymbol interference memory length is finite; thus, $r_{i-j} = 0$ for $|i - j| > m$.

In terms of this notation, the output z_i of the matched filter $m(t)$ can be written as

$$z_i = \sum_{j=-m+1}^{m-1} \hat{a}_{i+j} r_j + n_i. \quad (4)$$

The fact that the noise components n_k of the samples of the output of the matched filter are correlated $\{E(n_i n_{i+j}) = r_i\}$ was eliminated by the introduction of a whitening transversal filter [17] in early versions of the Viterbi algorithm (Forney [2], Kobayashi [18]). The whitening filter also shortens the intersymbol interference from two-sided in the past and future to one-sided involving only the past, and is equivalent to the forward filter of the decision feedback equalizer [19]. No loss of optimality is introduced by the whitening filter, but Underboeck [15] shows that a simple recursive calculation can also be obtained without the whitening filter. We shall use the nonwhitened approach because with (3), the optimal bit detector turns out to be simpler.

Notice that there is a recurrence in (3) in that the first two terms in the right-hand side are similar in form to the left-hand side. We shall continue this recurrence in the course of developing the algorithms. Notice also that the last three terms on the RHS of (3) are functions only of $\{\hat{a}_{N-m}, \hat{a}_{N-m+1}, \cdots, \hat{a}_N\}$ and not of the rest of the possible transmitted sequence. These two observations are at the core of the subsequent analysis.

We pause now to define some terms that are essential to the development of the technique. We define a set of state vectors

$$\sigma_k \triangleq \{\hat{a}_{k-m+1},\quad \hat{a}_{k-m+1}, \cdots, \hat{a}_k\};$$

$$k = m, m+1, \cdots, N \quad (5)$$

and define S^j to be the sequence of state vectors up to and including the state at time jT. Thus

$$S^k \triangleq \{\sigma_m, \sigma_{m+1}, \cdots, \sigma_k\} = \{\hat{a}_1, \hat{a}_2, \cdots, \hat{a}_k\}; \quad k \leqslant N.$$

Similarly, we define the sequence of outputs of the matched filter

$$z^k \triangleq \{z_1, z_2, \cdots, z_k\}; \qquad k \leqslant N.$$

We use the same notation in defining the sequences a^k and $\hat{a}^k$.

In terms of the elements in (3) we make the following definitions

$$U(z^k, S^k) \triangleq 2\sum_{i=1}^{k} \hat{a}_i z_i - \sum_{i=1}^{k}\sum_{j=1}^{k} \hat{a}_i \hat{a}_j r_{i-j} \quad (6a)$$

$$V(z_k, \sigma_{k-1}, \sigma_k) = 2\hat{a}_k z_k - 2\hat{a}_k \sum_{i=k-m}^{k-1} \hat{a}_i r_{k-i} + \hat{a}_k{}^2 r_0. \quad (6b)$$

From (6) and these definitions one can write the iterative relationship

$$U(z^k, S^k) = U(z^{k-1}, S^{k-1}) + V(z_k, \sigma_{k-1}, \sigma_k). \quad (7)$$

We return now to (2). Ignoring factors common to all sequences $\hat{a}_1, \hat{a}_2, \cdots, \hat{a}_N$ allows us to write

$$\Pr\,[z(t),\ \ 0 \leqslant t \leqslant \tau \,|\, a_1 = \hat{a}_1, a_2 = \hat{a}_2,\ \cdots,\ a_N = \hat{a}_N]$$
$$= c \exp U(z^N, s^N)$$
$$= c \exp\,[U(z^{N-1}, S^{N-1}) + V(z_N, \sigma_{N-1}, \sigma_N)] \qquad (8)$$

where c is a constant.

OPTIMUM SEQUENCE DETECTION

In optimum sequence detection we want to find the most probable transmitted sequence that resulted in a particular received signal. This is equivalent to maximum likelihood sequence estimation under the assumption that the transmitted symbols are independent and identically distributed. From (7) and (8) we have

$$\max_{\hat{a}^N} \Pr\,[a^N = \hat{a}^N \,|\, z(t), \qquad 0 \leqslant t \leqslant \tau]$$
$$= c \max_{\hat{a}^N} \exp\,[U(z^N, S^N)]. \qquad (9)$$

The problem of optimum sequence detection then comes down to maximizing $U(z^N, S^N)$ over all transmitted sequences.

From (7)

$$\max_{\hat{a}^N} U(z^N, S^N) = \max_{\sigma_m, \cdots, \sigma_N} [U(z^{N-1}, S^{N-1}) + V(z_N, \sigma_{N-1}, \sigma_N)]$$
$$= \max_{\sigma_N} \max_{\sigma_m, \cdots, \sigma_{N-1}|\sigma_N} [U(z^{N-1}, S^{N-1}) + V(z_N, \sigma_{N-1}, \sigma_N)]$$

where the notation $\sigma_m, \cdots, \sigma_{N-1} \,|\, \sigma_N$ means that σ_N is held fixed while $\sigma_m, \cdots, \sigma_{N-1}$ is varied. Continuing, we have

$$F(\sigma_N) \triangleq \max_{\sigma_m \cdots \sigma_{N-1}|\sigma_N} [U(z^{N-1}, S^{N-1}) + V(z_N, \sigma_{N-1}, \sigma_N)]$$
$$= \max_{\sigma_{N-1}|\sigma_N} \max_{\sigma_m \cdots \sigma_{N-2}|\sigma_{N-1}, \sigma_N} \cdot [U(z^{N-1}, S_{N-1}) + V(z_N, \sigma_{N-1}, \sigma_N)]. \qquad (10)$$

Two key observations allow us to proceed: 1) if σ_{N-1} and σ_N are fixed, $V(z_N, \sigma_{N-1}, \sigma_N)$ is independent of $\sigma_m, \cdots, \sigma_{N-2}$, and 2) $U(z^{N-1}, S^{N-1})$ is conditionally independent of σ_N, given σ_{N-1}. We have then

$$F(\sigma_N) = \max_{\sigma_{N-1}|\sigma_N} [V(z_N, \sigma_{N-1}, \sigma_N) + \max_{\sigma_m \cdots \sigma_{N-2}|\sigma_{N-1}} U(z^{N-1}, S^{N-1})]$$
$$= \max_{\sigma_{N-1}|\sigma_N} [V(z_N, \sigma_{N-1}, \sigma_N) + F(\sigma_{N-1})]. \qquad (11)$$

Notice that there is a recurrence in the term $F(\sigma_k)$. We have the general expression

$$F(\sigma_k) = \max_{\sigma_{k-1}|\sigma_k} [V(z_k, \sigma_{k-1}, \sigma_k) + F(\sigma_{k-1})]. \qquad (12)$$

By repeated application of (12), the optimum sequence can be found. The initial step of the calculation if $F(\sigma_m) \triangleq U(Z^m, S^m)$. At each step of the calculation, the path history leading to a particular value of a state is preserved. The maximization over the final set of states yields the optimum sequence.

We shall see that it is possible, through the merges, to make optimal decisions before the final set of states has been reached. At a given time there is associated with each state a number $F(\sigma_k)$, indicating the likelihood, and a path vector $P_j = \{a_1{*}, a_2{*}, \cdots a_2{*}, \cdots, a_{k-m}^*\}$, $j = 1, 2, \cdots, L^m$, indicating the states that were passed through in reaching the present state. The essence of the technique is that no other path gives a greater value of $F(\sigma_k)$. Now suppose that up to a certain point in the past all states have the same path. This is indicated by the fact that the first several components of each of the path vectors $P_j; j = 1, 2, \cdots, L^m$ are the same. This common path then must be a segment of the optimum sequence. This segment can be read out as a decision. It must be recognized that a merge is a random phenomenon and consequently may be of limited value in practical applications. (In applications, decisions which may be suboptimum are made after a certain maximum delay.) Experimental results show that this procedure does not lead to a precipitous decline in performance.

OPTIMUM SYMBOL-BY-SYMBOL DETECTION

In the previous section the whole received signal was used to detect the transmitted sequence. In optimal symbol-by-symbol detection, the whole received signal is used in the detection of each symbol separately. Again we have the received signal given by (1), and we assume the channel has finite memory as expressed in (4). In the sequel we shall focus on the detection of one of the N transmitted symbols, which we designate as a_ω, where $1 \leqslant \omega \leqslant N$.

The detection process consists of attempting to find the most likely value of a_ω given the received signal, i.e., we find $\hat{a}_\omega$ such that $\Pr\,[a_\omega = \hat{a}_\omega \,|\, z(t), 0 < t < \tau]$ is maximized. In order to proceed with the derivation of a tractable expression for this likelihood, we define a subset of the set of states [see (5)]. Let $\tilde{\sigma}_{jl}; j = 1, 2, \cdots, N; l = 1, 2, \cdots, L$ denote the same set of states as defined previously except that $\hat{a}_\omega$ is set equal to one of its L possible values, i.e., $\hat{a}_\omega = l$. Thus $\tilde{\sigma}_{jl} = \sigma_j$, for $j \leqslant \omega$ and $j \geqslant \omega + m$. For $\omega \leqslant j < \omega + m$, $\tilde{\sigma}_{jl}$ has L possible values depending on the L possible values of $\hat{a}_\omega$. We also define $\tilde{S}_l^k = \{\tilde{\sigma}_{ml}, \tilde{\sigma}_{m+1,l}, \cdots, \tilde{\sigma}_{kl}\}$.

Armed with these definitions, we proceed to find

$$\Pr[a_\omega = \hat{a}_\omega \mid z(t), \quad 0 \leqslant t \leqslant \tau] = \sum_{\tilde{S}_l^N} \frac{\Pr[a_1 = \hat{a}_1, \cdots, a_N = \hat{a}_N] \Pr[z(t),\ 0 \leqslant t \leqslant \tau \mid a_1 = \hat{a}_1, \cdots, a_N = \hat{a}_N]}{\Pr[z(t),\ 0 \leqslant t \leqslant \tau]}. \tag{13}$$

We recognize that $\Pr[a_1 = \hat{a}_1, \cdots, a_N = \hat{a}_N] = L^{-N}$ and that $\Pr[z(t), 0 \leqslant t \leqslant \tau]$ is common to all terms. From (8) and (9) we have

$$\max_{\hat{a}_\omega} \Pr[a_\omega = \hat{a}_\omega \mid z(t), \quad 0 \leqslant t \leqslant \tau] = c \max_{\hat{a}_\omega} \sum_{\tilde{\sigma}_{ml} \cdots \tilde{\sigma}_{Nl}} \exp[U(z^N, \tilde{S}^N)]. \tag{14}$$

In order to calculate the summation in (14), we use a similar recurrence to that used previously [see (8)]:

$$\sum_{\tilde{\sigma}_{ml} \cdots \tilde{\sigma}_{Nl}} \exp[U(z^N, S^N)] = \sum_{\tilde{\sigma}_{Nl}} \left(\sum_{\tilde{\sigma}_{N-1,l} | \tilde{\sigma}_{Nl}} \left(\sum_{\tilde{\sigma}_{ml} \cdots \tilde{\sigma}_{N-2,l} | \tilde{\sigma}_{N-1,l} \tilde{\sigma}_{Nl}} (\exp[U(z^{N-1}, S^{N-1}) + V(z_N, \tilde{\sigma}_{N-1,l}, \tilde{\sigma}_{Nl})]) \right) \right)$$

where

$$\sum_{\tilde{\sigma}_{N-1,l} | \tilde{\sigma}_{Nl}}$$

denotes summation over all realizations of the state $\tilde{\sigma}_{N-1,l}$ holding state $\tilde{\sigma}_{Nl}$ fixed. Similarly,

$$\sum_{\tilde{\sigma}_{ml} \cdots \tilde{\sigma}_{N-2,l} | \tilde{\sigma}_{N-1,l} \tilde{\sigma}_{Nl}}$$

denotes summation over $\tilde{\sigma}_{ml} \cdots \tilde{\sigma}_{N-2,l}$ holding $\tilde{\sigma}_{N-1,l}, \tilde{\sigma}_{Nl}$ fixed. By the same line of reasoning which led to (11) we have

$$\sum_{\tilde{\sigma}_{ml} \cdots \tilde{\sigma}_{Nl}} \exp[U(z^N, S^N)] = \sum_{\tilde{\sigma}_{Nl}} \left(\sum_{\tilde{\sigma}_{N-1,l} | \tilde{\sigma}_{Nl}} \exp[V(z_N, \tilde{\sigma}_{N-1,l}, \tilde{\sigma}_{Nl})] \cdot \left(\sum_{\tilde{\sigma}_{ml} \cdots \tilde{\sigma}_{N-2,l} | \tilde{\sigma}_{N-1,l}} \cdot \exp[U(z^{N-1}, \tilde{S}_l^{N-1})] \right) \right). \tag{15}$$

By defining the expressions

$$G(\tilde{\sigma}_{kl}, \hat{a}_\omega) = \sum_{\tilde{\sigma}_{ml} \cdots \tilde{\sigma}_{k-1,l} | \tilde{\sigma}_{kl}} \exp[U(z^k, \tilde{S}_l^k)] \qquad k = m+1, \cdots, N$$

and

$$G(\tilde{\sigma}_{ml}, \hat{a}_\omega) = \exp[U(z^m, \tilde{S}_l^m)],$$

we can write

$$G(\tilde{\sigma}_{kl}, \hat{a}_\omega) = \sum_{\tilde{\sigma}_{k-1,l} | \tilde{\sigma}_{kl}} \exp[V(z_k, \tilde{\sigma}_{k-1,l}, \tilde{\sigma}_{kl})] G(\tilde{\sigma}_{k-1,l}, \hat{a}_\omega); \qquad k = m+1, m+2, \cdots, N. \tag{16}$$

The steps required to find the optimum value of $\hat{a}_\omega$ can be summarized as follows. For each of the L possible values of $\hat{a}_\omega$, we make the following computations:

1) Compute for each $\tilde{\sigma}_{ml}$ the quantity $G(\tilde{\sigma}_{ml}, \hat{a}_\omega)$.

2) Using the iterative relationship (16), compute in succession the quantities $G(\tilde{\sigma}_{m+1,l}, \hat{a}_\omega)$, $G(\tilde{\sigma}_{m+2,l}, \hat{a}_\omega) \cdots G(\tilde{\sigma}_{Nl}, \hat{a}_\omega)$.

3) Sum $G(\tilde{\sigma}_{Nl}, \hat{a}_\omega)$ over all states $\tilde{\sigma}_{Nl}$ and choose $\hat{a}_\omega$ which produces a maximum.

As in optimum sequence detection, the finite memory of the channel allows us to define a suitable set of states, thereby avoiding complexity that grows exponentially with the length of the transmitted sequence. However, the foregoing calculation must be repeated for each of the N transmitted symbols, implying an N^2 growth in complexity.

There is a degree of commonality in the calculations for each of the symbols, which may reduce complexity to some extent. Consider, for example, the detection of symbols $a_{\omega 1}$ and $a_{\omega 2}$ where $1 \leqslant \omega_1 \leqslant \omega_2 \leqslant N$. For $j < \omega_1$, the state $\tilde{\sigma}_{jl}$ is independent of $a_{\omega 1}$ and $a_{\omega 2}$ and we have $G(\tilde{\sigma}_{jl}, \hat{a}_{\omega 1}) = G(\tilde{\sigma}_{jl}, \hat{a}_{\omega 2})$. Thus, in order to detect the symbols $\hat{a}_{\omega 1}, \hat{a}_{\omega 1} + 1, \cdots, \hat{a}_N$, L^m values of $G(\tilde{\sigma}_{jl}, \hat{a}_{\omega 1})$ need be computed for the states $\tilde{\sigma}_{jl}$, $j = 1, 2, \cdots, \omega_1 - 1$. When at the ω_1 th step, a distinction must be made among the possible values of $\hat{a}_{\omega 1}$. Until a decision is made on $\hat{a}_{\omega 1}$, the quantities $G(\tilde{\sigma}_{jl}, \hat{a}_{\omega 1})$ must be computed for all possible values of $\hat{a}_{\omega 1}$. In carrying out the computations for the successive states, we carry along $G(\tilde{\sigma}_{jl}, \hat{a}_N)$ which is used in the calculations of $G(\tilde{\sigma}_{jl}, \hat{a}_k)$; $j, k < N$. This commonality reduces the complexity of computation by one-half.

MERGES

A property that is analogous to merges in optimum sequence detection may offer the possibility of further reductions in complexity. Let $\{\alpha_1, \alpha_2, \cdots, \alpha_L\}$ denote the values that a symbol a_ω; $1 \leqslant \omega \leqslant N$ may assume. As indicated in (14), it is decided that $a_\omega = \alpha_l$ if

$$\sum_{\tilde{\sigma}_{Nl}} G(\tilde{\sigma}_{Nl}, \hat{a}_\omega)|_{\hat{a}_\omega = \alpha_l} \geqslant \sum_{\tilde{\sigma}_{Nl}} G(\tilde{\sigma}_{Nl}, \hat{a}_\omega)|_{\hat{a}_\omega \neq \alpha_l}. \tag{17}$$

Now suppose that for a particular j such that $j \geqslant m + \omega$

$$G(\tilde{\sigma}_{jl}, \hat{a}_\omega)|_{\hat{a}_\omega = \alpha_l} \geqslant G(\tilde{\sigma}_{jl}, \hat{a}_\omega)|_{\hat{a}_\omega \neq \alpha_l} \quad \forall \tilde{\sigma}_{jl}. \tag{18}$$

From (15) it follows that $\forall k \geqslant j$:

$$G(\tilde{\sigma}_{kl}, \hat{a}_\omega)|_{\hat{a}_\omega = \alpha_l} \geqslant G(\tilde{\sigma}_{jl}, \hat{a}_\omega)|_{\hat{a}_\omega \neq \alpha_l} \quad \forall \tilde{\sigma}_{kl}. \tag{19}$$

Consequently, (17) holds.

Thus, if the relationship in (18) holds, then it is not necessary to compute $G(\tilde{\sigma}_{kl}, \hat{a}_\omega)$; $k > j$. The decision $\hat{a}_\omega = \alpha_l$ can be made at time jT. In analogy with the Viterbi algorithm, a practical compromise would be to force a decision after a certain delay.

IMPLEMENTATION OF ML SEQUENCE DETECTION AND SYMBOL-BY-SYMBOL DETECTION

The calculations that must be carried out in each symbol interval for the algorithms under consideration are indicated for sequence detection in (12), and for symbol-by-symbol detection in (16). We begin the comparison of relative complexity by noting that the quantity $V(Z_k, \sigma_{k-1}, \sigma_k)$ [see (6)] is common to both. Ostensibly, (6) indicates $m + 3$ multiplications and $m + 2$ additions. However, this is far too pessimistic. The quantities $\hat{a}_k \sum_{i=k-m}^{k-1} \hat{a}_i r_{k-i} + a_k^2 r_0$ may be computed one time and stored. Nor is it necessary to store all possible values of these quantities. The α_i; $i = 1, 2, \cdots, L$ are assumed to be regularly spaced integer values. Therefore, we can increment through the whole range of states starting from one particular state simply by additions. The same can be said for the calculation of the quality $\hat{a}_k z_k$. In the sequel we denote the number of additions necessary to compute $V(Z_k, \sigma_{k-1}, \sigma_k)$ for a single pair of states as equivalent to A_V additions.

From (12) we see that in order to find $F(\sigma_k)$ for a single realization of σ_k, $L(A_V + 1)$ additions and $L - 1$ comparisons are necessary. Thus, for all realizations of the state σ_k, L^{m+1} $(A_V + 1)$ additions and $L^m(L-1)$ comparisons must be carried out in each symbol interval. If we assume that a comparison is equivalent to an addition, we have for the total number of equivalent additions

$$S_1 = L^m(LA_V + 2L - 1). \tag{20}$$

Memory must also be considered in assessing the relative complexity of the two techniques. The optimum sequence estimator carries along a path history and the quantity $F(\sigma_k)$ for each state from symbol interval to symbol interval. Let R_1 denote the number of bits necessary to store $F(\sigma_k)$ with sufficient accuracy. We also assume that the path history is allowed to reach some maximum length M_1 whereupon decisions are made. This approach is suboptimum. However, if M_1 is large enough, a significant number of merges will take place before the path length is truncated. We find then that the storage requirement for optimum sequence detection is

$$B_1 = L^m[R_1 + M_1 \log_2 L] \text{ bits}. \tag{21}$$

Here B_1 is a minimum value since we have not taken into account any memory required to calculate $V(Z_k, \sigma_{k-1}, \sigma_k)$.

The estimation of the complexity of optimum symbol-by-symbol detection is more complicated. Recall that the quantity $G(\tilde{\sigma}_{jl}, \hat{a}_N)$ is computed in each symbol interval. Now from (16) we see that this requires $LA_V + L-1$ additions, L multiplications, and L exponentiations for each state. We shall assume that a single multiplication is equivalent to P additions. There are several ways that the exponential may be calculated. The most costly is to compute the series expansions term-by-term, implying a number of multiplications and additions. On the other hand, the use of read-only memory may reduce the number of operations that are required. Let us denote the number of additions that are equivalent to exponentiation as E. The total number of equivalent additions to calculate $G(\tilde{\sigma}_{jl}, \hat{a}_N)$ for all states is then $L^{m+1}(A_V + P + E + 1) - L^m$ additions in each symbol interval. These are not all of the calculations that must be performed. For each $G(\tilde{\sigma}_{jl}, \hat{a}_k)$, $k \leqslant j$, a separate set of calculations must be performed for each possible value of $\hat{a}_k$. It is possible that these must be done for all $j = k, k+1, \cdots, N$. However, we shall truncate as in optimum sequence detection. We assume that after M_2 symbol intervals a decision on a transmitted symbol is made. The total number of equivalent additions for optimum bit-by-bit detection is

$$\begin{aligned} S_2 &= L^m(LA_V + PL + EL + L - 1) \\ &\quad + M_2 L^{m+1}(LA_V + PL + EL + L - 1) \\ &= L^m(LA_V + PL + EL + L - 1)(1 + M_2 L). \end{aligned} \tag{22}$$

The amount of storage required for bit-by-bit detection is governed by the fact that the quantities $G(\tilde{\sigma}_{jl}, \hat{a}_k)$ must be stored for each state, for each possible value of the M_2 undetected symbols. We assume that R_2 bits are required to store the quantity $G(\tilde{\sigma}_{jl}, \hat{a}_k)$ with sufficient accuracy. We find that the required memory is

$$B_2 = L^m R_2 [1 + M_2 L]. \tag{23}$$

CONCLUSION

We have derived an algorithm for optimum symbol-by-symbol detection. The starting point for the derivation is the Viterbi algorithm used for optimum sequence detection. The difference in the optimality criterion can be brought into focus by the following considerations. Let $\boldsymbol{a}$ be a random finite length sequence of ±1's. The Viterbi algorithm finds the vector $\hat{a}$ which minimizes $P(\hat{a} \neq a)$. The symbol-by-symbol algorithm in this paper finds the vector $\hat{a}$ which minimizes the expected number of places in which $\hat{a}$ and a disagree. Thus, the first algorithm minimizes the probability of any error whatsoever, and the second minimizes the expected number of symbol errors. Both algorithms exhibit the merging phenomenon. That is, a sequential computation allows the fixing of a certain number of a_i's at the beginning of the sequence whenever it can be shown that no further received information of any kind will overturn the decision on this initial segment. The merge phenomenon leads to a lower complexity in the computation of $\hat{a}$.

REFERENCES

[1] A. J. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Trans. Inform. Theory*, vol. IT-13, pp. 260–269, Apr. 1967.

[2] G. D. Forney, Jr., "Maximum-likelihood sequence estimation of digital sequences in the presence of intersymbol interference," *IEEE Trans. Inform. Theory*, vol. IT-18, pp. 363–478, May 1972.

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

[4] K. Abend and B. D. Fritchman, "Statistical detection for communication channels with intersymbol interference," *Proc. IEEE*. vol. 58, pp. 779–785, May 1970.

[5] R. W. Chang and J. C. Hancock, "On receiver structures for channels having memory," *IEEE Trans. Inform. Theory*, vol. IT-12, pp. 463–468, Oct. 1966.

[6] S. U. H. Qureshi and E. E. Newhall, "An adaptive receiver for data transmission over time-dispersive channels," *IEEE Trans. Inform. Theory*, vol. IT-19, pp. 448–451, July 1973.

[7] D. D. Falconer and F. R. Magee, Jr., "Adaptive channel memory truncation for maximum likelihood sequence estimation," *Bell Syst. Tech. J.*, vol. 52, pp. 1541–1561, Nov. 1973.

[8] F. L. Vermeulen and M. E. Hellman, "Reduced state Viterbi decoding for channels with intersymbol interference" (abstract), presented at the IEEE Int. Symp. Inform. Theory, 1972.

[9] G. J. Foschini, "A reduced state variant of maximum likelihood sequence detection attaining optimum performance at high signal-to-noise ratios," *IEEE Trans. Inform. Theory*, vol. IT-23, pp. 605-609, Sept. 1977.

[10] P. J. McLane, "A residual intersymbol interference error bound for truncated state Viterbi detectors," *IEEE Trans. Inform. Theory*, vol. IT-26, pp. 548–553, Sept. 1980.

[11] M. Wilson *et al.*, "Microprocessor implementation of the Viterbi detector," in *Proc. Nat. Telecommun. Conf.*, Washington, DC, Nov. 1979.

[12] B. Ross *et al.*, "Microprocessor realization of the adaptive Viterbi detector," presented at the Nat. Telecommun. Conf., Houston, TX, Dec. 1980.

[13] R. S.-W. Cheung and P. J. McLane, "Carrier reference error sensitivity of Viterbi detector for PAM data transmission," *IEEE Trans. Commun.*, to be published.

[14] R. W. Lucky, J. Salz, and E. J. Weldon, Jr., *Principles of Data Communications*. New York: McGraw-Hill, 1968, ch. 6.

[15] G. Ungerboeck, "Adaptive maximum-likelihood receiver for carrier modulated data-transmission systems." *IEEE Trans. Commun.*, vol. COM-22, pp. 624–636, May 1974.

[16] L. K. Mackechnie, "Receivers for channels with intersymbol interference," (Abstract), presented at the IEEE Int. Symp. Inform. Theory, 1972.

[17] I. N. Andersen, "Sample whitened matched filters," *IEEE Trans. Inform. Theory*, vol. IT-19, pp. 653–660, Sept. 1973.

[18] H. Kobayashi, "Correlative level coding and maximum likelihood decoding," *IEEE Trans. Inform. Theory*, vol. IT-17, pp. 586–594, Sept. 1971.

[19] R. Price, "Nonlinearly feedback-equalized PAM vs. capacity, for noisy filter channels," in *Proc. Int. Conf. Commun.*, 1972, pp. 22-12–22-18.

[20] J. F. Hayes, "The Viterbi algorithm applied to digital data transmission," *Commun. Soc.*, vol. 13, pp. 15–20, Mar. 1975.

Jeremiah F. Hayes (S'65–M'66–SM'79) received the B.E.E. degree from Manhattan College, New York, NY, in 1956, the M.S. degree in mathematics from New York University, New York, in 1961, and the Ph.D. degree in electrical engineering from the University of California, Berkeley, in 1966.

From 1956 to 1960 he was a member of the Technical Staff at Bell Laboratories, Murray Hill, NJ. He worked at the Columbia University Electronics Research Laboratories, New York, from 1960 to 1962. In the interval 1966–1969 he was a member of the faculty at Purdue University, Lafayette, IN. During the summer of 1967 he was employed by the Jet Propulsion Laboratory, Pasadena, CA. From 1969 to 1978 he was a member of the Technical Staff at Bell Laboratories, Holmdel, NJ. Since September 1978 he has been a member of the Department of Electrical Engineering, McGill University, Montreal, P.Q., Canada. His research interest is primarily in the area of computer communications.

Dr. Hayes is currently the Editor for Computer Communication of this TRANSACTIONS.

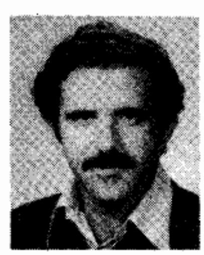
★

Thomas M. Cover (S'59–M'66–SM'72–F'74) was born in San Bernardino, CA, on August 7, 1938. He received the B.S. degree in physics from the Massachusetts Institute of Technology, Cambridge, in 1960, and the M.S. and Ph.D. degrees, both in electrical engineering, from Stanford University, Stanford, CA, in 1961 and 1964, respectively.

He is currently a Professor in the Departments of Electrical Engineering and Statistics at Stanford University, where he teaches and does research in the areas of information theory, pattern recognition, and complexity. He has been on the faculty at Stanford since 1964, and has served as a consultant to SRI, Bell Laboratories, and Sylvania. He served in the Visiting Lecturer Program in Statistics from 1971 to 1974. In the academic year 1971–1972, he was a Visiting Associate Professor in Electrical Engineering at M.I.T. and a Vinton Hayes Research Fellow at Harvard University, Cambridge.

Dr. Cover was the Chairman of the IEEE Information Theory Group in 1972, Book Review Editor for the IEEE TRANSACTIONS ON INFORMATION THEORY, and received the 1972–1973 Information Theory Paper Award. He is currently a member of the Information Theory Group Board of Governors. He is a reviewer for *Math Reviews* and is a member of the Standing Committee for the International Association for Pattern Recognition. He is a member of IMS, AMS, ACM, SIAM, IAPR, URSI, and Sigma Xi.

★



Juan B. Riera (S'70–M'72) was born in Valencia, Spain, in December 1946. He received the Engineer of Telecommunications degree and the Ph.D. degree from the Universidad Politecnica de Madrid, Madrid, Spain, in 1969 and 1973, respectively.

Since 1969 he has been a Faculty Member of the Universidad Politecnica. In 1973 and 1975 he was on leave in the U.S. at Stanford University, Stanford, CA. He is currently Professor at the Telecommunications School and Faculty of Informatics at the Universidad Politecnica de Madrid where he teaches and does research in the areas of data transmission and computer networks.