Convolutional Codes
III. Sequential Decoding*

G. DAVID FORNEY, JR.

Codex Corporation, Newton, Massachusetts 02195

Sequential decoding is characterized as a sequential search for the shortest path through a trellis. An easily analyzed algorithm closely related to the stack and Fano algorithms is described. Martingale techniques are used to find the distribution of computation on totally symmetric channels. For general channels, our universal bounding technique yields the well-known Pareto distribution of computation, as well as a bound on error probability that is asymptotically optimum in the high-rate range. The performance-complexity relation is shown to be asymptotically Pareto for both sequential and maximum-likelihood (Viterbi) decoding, with the same exponent in either case. A semi-sequential list-of-$L$ Viterbi algorithm is introduced to extend the analogies below $R_{\text{comp}}$.

I. INTRODUCTION

In Part II (Forney, 1974) maximum-likelihood decoding was characterized as the determination of the shortest path through a topological structure called a trellis. We discussed an efficient algorithm (the Viterbi algorithm) that systematically examines all possible paths. Heuristically, one would suspect that most of the time the correct path could be accurately estimated just by looking a few branches ahead at a time, since it would tend to be much shorter than all incorrect paths diverging from it, and that therefore most of the computation in the Viterbi algorithm could be avoided. This notion is at the heart of various algorithms generically known as sequential decoding.

In this paper we first describe the basic features of sequential algorithms: first, incremental search for the correct path by extension of previously examined paths; second, adoption of a biased "metric" to make the search efficient. We then describe the pedagogically clean algorithm we use for our

* This work was partially supported by NSF Grant GK-31627 and by the Industrial Affiliates Program in Information Systems at Stanford University.
derivations; later we indicate how to extend the results to more practical algorithms.

The main theorems for sequential decoding have to do not only with error probability but also, since the algorithms involve random searches, the distribution of the random variable of searching effort or "computation." In Appendix A, we consider separately the characteristics of the correct path and of the incorrect subset. In the main text, we develop from first principles the main sequential decoding theorem, namely that the probability of more than \( N \) computations decreases as \( N^{\rho_r} \) (the Pareto distribution). Finally, we show that in the range \( 0 \leq \rho_r \leq 1 \) the error probability is asymptotically the same as that of maximum-likelihood decoding.

Sequential decoding actually has a much longer history than maximum-likelihood decoding of convolutional codes, and all the main results have been developed in an isolated and frequently difficult literature. We hope that the unified consideration of the two techniques in these two papers will broaden their accessibility to the uninitiated, as well as give the specialist a useful joint perspective.

### II. Basic Principles

Given the problem of starting at a given node of a graph and finding the shortest path to another node, where the graph topology is known but the length of each branch unknown, it would seem sensible to proceed generally as follows. Start computing path lengths from the origin node out, keeping track of the total length out to the last branch examined. Select new branches to be examined from those which extend the shorter paths discovered so far, the specific method depending on the algorithm. Evidently the final node will be reached sooner or later along one of the shortest paths without any time being spent extending paths much longer than the shortest overall path.

Since this heuristic lies at the heart of all sequential decoding methods, we shall follow Jacobs and Berlekamp (1967) by characterizing a sequential decoding algorithm as follows.

**Definition 1.** A decoder is **sequential** only if it computes a subset of path lengths in a sequential fashion, each new path being an extension of a previously examined one, and the decision as to which path to extend being based only on already examined paths.

We next observe that in a very long trellis the total length of even the best path becomes very large, so that to execute the above procedure we must be
continually going back and extending early paths, even those that seem very bad and unlikely to be headed anywhere. The second fundamental idea of sequential decoding is to bias the path lengths so that this does not happen. We recall that the "length" corresponding to a single channel symbol was defined as $-\ln p_{jk}$, where $p_{jk} = \Pr(j \mid k)$ is the probability of observing channel output $j$, given the channel input symbol $k$ that corresponds to the given branch at that time. Observe that we can add any quantity $f_i$ to all lengths at time $i$, where $f_i$ may differ for different times $i$, without disturbing the relative ranking of lengths of different paths between any node and any other, since all such paths go through the same set of times $i$. The bias that turns out to minimize computation is $f_i = \ln q_j + r$, where $q_j = \sum_k p_k p_{jk}$ is the ensemble average probability of the output $j$ actually observed at time $i$. [For a discussion of the effects of bias see Jelinek (1974).] The sign is also usually inverted, so that the metric increment is defined as

$$m_{jk} = \ln \frac{p_{jk}}{q_j} - r,$$

and the objective becomes the finding of the path with the maximum accumulated metric. We see subsequently that with this bias the correct path tends to have an increasing metric, while incorrect paths decrease, and more importantly the best path in any incorrect subset tends to decrease. These are the minimum requirements for any of the known sequential decoding algorithms to work efficiently.

### III. A STACK ALGORITHM

We now introduce a particular algorithm, which closely resembles the Zigangirov (1966)–Jelinek (1969) algorithm, but takes account of merging as the Viterbi maximum-likelihood algorithm does. The algorithm at any time stores a "stack" of $N$ partial paths $y^{(n)}$ ordered according to their metrics.

**Algorithm A**

**Initialization.** $N = 1$, $y^{(1)}$ = zero-length path (origin node).

**Recursion.** Compute the metrics of the $M$ single-branch continuations of the best path $y^{(1)}$. If any such path merges with a path already in the stack, discard the worse one of the pair. Reorder the remaining paths. If the best path $y^{(1)}$ now terminates in the final node, end; otherwise, repeat.
Refinement 1 (Algorithm Z). Eliminate all paths with metrics less than the best path metric minus some parameter $\beta$.

Coarsening 1 (Algorithm ZJ). Forget about eliminations due to merging.

Coarsening 2 (Algorithm J). In the ordering operation, sort all paths with metrics in the same quantization interval $kA \leqslant \text{metric} < (k + 1)A$ into bin $k$ in arbitrary order, where $A$ is some parameter and $k$ ranges over the integers.

We now state some elementary properties of Algorithm A that will be the basis for our study of the distribution of computation. We use the notation $y_t$ for the correct path to time $t$, and $\Gamma(y_t)$ for its metric (at time $t$). We denote by $\gamma_t$ the size of the dip from $\Gamma(y_t)$ to the lowest subsequent metric on the correct path:

$$\gamma_t = \Gamma(y_t) - \min_{r \geq t} \Gamma(y_r).$$

Thus $\gamma_t \geq 0$. We define the number of time-$t$ computations $C_t$ as the total number of paths $y'$ in the time-$t$ incorrect subset $S_t$ that are ever extended by Algorithm A.

**Lemma 1.** Let $y$ and $y'$ be two paths springing from some common node, and let the minimum metric on the path $y$ after the common node exceed $\Gamma(y')$. Then Algorithm A cannot extend $y'$ before extending $y$, or some path that has merged with $y$.

**Proof.** To extend either, the common node must first be reached. At all subsequent times before $y$ is extended, the stack contains a path which is a truncation of $y$ at the common node or later; or else it contains some path which has merged with $y$ with higher metric. By the hypothesis there is therefore always a path of metric exceeding $\Gamma(y')$ from the time the common node is reached to the time $y$ or a path merged with $y$ is extended, hence $y'$ cannot be extended during this time. Q.E.D.

**Theorem 1.** The number of time-$t$ computations $C_t$ is overbounded by the number of paths $y'$ in the time-$t$ incorrect subset $S_t$ with metrics not less than the minimum metric on the correct path at time $t$ or later:

$$C_t \leq \sum_{y' \in S_t} \phi(y'),$$

where

$$\phi(y') = \begin{cases} 1, & \text{if } \Gamma(y') \geq \Gamma(y_t) - \gamma_t = \min_{r \geq t} \Gamma(y_r); \\ 0, & \text{otherwise.} \end{cases}$$

(4)
Proof. We suppose $y_t$ is reached, else $C_t = 0$. By Lemma 1, if $\Gamma(y') < \Gamma(y_t) - \gamma_t$, there is always some truncation of the correct path or a path merged with the correct path in the stack with metric not less than $\Gamma(y_t) - y_t$, and $y'$ can never be extended. Q.E.D.

Now we make similar observations concerning error probability. We define a (time-\(t\)) proto-error as the event in which some path (possible error event) $y'$ in the time-\(t\) incorrect subset $S_t$ has a metric $\Gamma(y')$ at the point of merger with the correct path that equals or exceeds the minimum metric $\Gamma(y_t) - \gamma_t$ on the correct path at time $t$ or later (whether before or after the point of merger). By Lemma 1, $y'$ will never be extended unless $\Gamma(y') \geq \Gamma(y_t) - \gamma_t$, hence no error can occur without a proto-error (the converse is of course not true). We can therefore bound the error probability by the probability of proto-error, which will turn out to be asymptotically tight.

IV. Computational Distribution

In this section, we shall derive the general result that $\Pr(C_t \geq N) \leq K N^{-p}$ for any $p < p_r$, for the two cases $0 < p \leq 1$ and $1 < p \leq 2$. The techniques used are basically those of Appendix A of Part II, with considerable parallels to the list-of-1 and list-of-2 error probability derivations.

As in Part II, we choose a random $(M, v, n)$ trellis code. However, it is a nuisance to have to consider merging; hence we observe that the number of nodes $C_t$ above the threshold $\Gamma(y_t) - \gamma_t$ is overbounded by the number $C'_t$ of nodes in a tree in which nodes which would otherwise be eliminated by merging are allowed to continue branching with the same statistics; or equivalently the number of nodes $y' \in S_t'$, where $S_t'$ is the time-\(t\) incorrect subset of an $(M, \infty, n)$ code of infinite constraint length, in which there is no merging. We shall henceforward, as has always been done previously in the literature, bound the distribution of $C_t'$, while noting that in Algorithm A $C_t'$ is a true upper bound on $C_t$.

Theorem 1 shows that $C_t'$ is bounded by the number of incorrect paths $y'$ in the time-\(t\) incorrect subset $S_t'$ (of an infinite-constraint-length code), such that

$$\Gamma(y') \geq \Gamma(y_t) - \gamma_t = \min_{\tau \geq t} \Gamma(y_{\tau}),$$

where $y_{\tau}$ is the correct path to time $\tau$. For notational convenience, we shall henceforth let the metric $\Gamma(y_t)$ of the starting node be 0, and let the starting time $t$ be 0.
For step 2, we first consider the quantity

\[ T(\alpha, \rho) = \left[ \sum_{y \in S_0} e^{\alpha \Gamma(y')} \right]^\rho, \quad \alpha \geq 0. \]  

(6)

If there are more than \( N \) nodes with \( \Gamma(y') \geq \Gamma(y_0) - \gamma_0 = -\gamma_0 \), then

\[ T(\alpha, \rho) \geq N^\rho \exp -\alpha \rho \gamma_0. \]  

(7)

Hence

\[ \Pr(C_0' \geq N) \leq N^{-\rho} \left[ \exp \alpha \rho \gamma_0 \right] T(\alpha, \rho). \]  

(8)

For \( 1 < \rho \leq 2 \), we will want to rewrite \( T(\alpha, \rho) \) as

\[ T(\alpha, \rho) = \left[ \sum_{y_1 \in S_0} \sum_{y_2 \in S_0'} e^{2\Gamma(y_1')} e^{2\Gamma(y_2')} \right]^{\rho/2}. \]  

(9)

Finally, we bound the correct-path term by

\[ \exp \alpha \rho \gamma_0 = \exp -\alpha \rho \left[ \min_{t \geq 0} \Gamma(y_t) \right] \leq \sum_{t=0}^{\infty} \exp -\alpha \rho \Gamma(y_t). \]  

(10)

The combination of (6) or (9) with (8) and (10) gives us bounds in the desired form:

\[ \Pr(C_0' \geq N) \leq N^{-\rho} \sum_{t=0}^{\infty} \exp -\alpha \rho \Gamma(y_t) \left[ \sum_{y' \in S_0'} \exp \alpha \Gamma(y') \right]^\rho, \quad \alpha \geq 0, \quad 0 \leq \rho \leq 1; \]

\[ \leq N^{-\rho} \sum_{t=0}^{\infty} \exp -\alpha \rho \Gamma(y_t) \left[ \sum_{y_1' \in S_0'} \sum_{y_2' \in S_0'} \exp \alpha(\Gamma(y_1') + \Gamma(y_2')) \right]^{\rho/2}, \quad \alpha \geq 0, \quad 0 \leq \rho \leq 2. \]  

(11)

Next, configuration-counting. We now have to deal with configurations of a truncation \( y_t \) of the correct path, as well as truncations of incorrect paths \( y' \) in an infinite-constraint-length tree. Let \( C_{t\tau} \) be the set of all pairs \( (y_t, y_{\tau'}) \), where \( y_t \) is the correct path to time \( t \), and \( y_{\tau'} \) any incorrect path in \( S_0' \) to time \( \tau \). Clearly \( |C_{t\tau}| \leq M' \). Further let \( C_{t\tau_{\gamma_0,\gamma_1}} \) be the set of all triples
(y_t, y'_t, y''_t) in which y'_t and y''_t are merged for \( \tau_0 \) time units; clearly

\[
\left| C_{t_00,t_02} \right| \leq M^r e^M e^M e^{-M^r_{t_0}}.
\]

We now use Jensen’s inequality to rewrite (11) as

\[
\Pr(C_{t'} \geq N) \leq N^{-\rho} \sum_{l=0}^{\infty} \left[ \sum_{\tau=0}^{\infty} \sum_{(y_t, y'_t) \in C_{t'}} e^{a\Gamma(y'_t)} e^{-a\Gamma(y_t)} \right]^\rho
\]

\[
\leq N^{-\rho} \sum_{l=0}^{\infty} \sum_{\tau=0}^{\infty} \left[ \sum_{(y_t, y'_t) \in C_{t'}} e^{a\Gamma(y'_t)} e^{-a\Gamma(y_t)} \right]^\rho, \quad 0 \leq \rho \leq 1,
\]

\[
= N^{-\rho} \sum_{l=0}^{\infty} \sum_{\tau=0}^{\infty} T_{t'\tau}(x, \rho);
\]

\[
\Pr(C_{t'} \geq N) \leq N^{-\rho} \sum_{l=0}^{\infty} \sum_{\tau=0}^{\infty} \sum_{\kappa=0}^{\infty} \sum_{\zeta=0}^{\infty} \sum_{(y_t, y'_t) \in C_{t0}\tau_0^0} e^{a\Gamma(y'_t)} e^{a\Gamma(y_t)} e^{-2a\Gamma(y_t)} \left[ \sum_{(y_t, y'_t, y''_t) \in C_{t0}\tau_0^0} \sum_{(y_t, y'_t) \in C_{t0}\tau_0^0} \sum_{(y_t, y'_t) \in C_{t0}\tau_0^0} \sum_{(y_t, y'_t) \in C_{t0}\tau_0^0} T_{t0\tau_0^0\tau_0^0}(x, \rho) \right]^{\rho/2},
\]

\[
\leq N^{-\rho} \sum_{l=0}^{\infty} \sum_{\tau=0}^{\infty} \sum_{\kappa=0}^{\infty} \sum_{\zeta=0}^{\infty} T_{t0\tau_0^0\tau_0^0}(x, \rho), \quad 0 \leq \rho \leq 2;
\]

\[
\leq N^{-\rho} \sum_{l=0}^{\infty} \sum_{\tau=0}^{\infty} \sum_{\kappa=0}^{\infty} \sum_{\zeta=0}^{\infty} T_{t0\tau_0^0\tau_0^0}(x, \rho), \quad \text{(13)}
\]

where \( T_{t0}(x, \rho) \) and \( T_{t0\tau_0^0\tau_0^0}(x, \rho) \) are defined implicitly. This completes Step 3.

Now we apply Lemma A3 of Part II. For the first bound, there are three time-slice types, described in Table I. Here \( \mu_i \) is the number of incorrect paths in slice \( l \).

Hence Lemma A3 gives, for \( 0 \leq \rho \leq 1 \),

\[
T_{t0}(x, \rho) \leq |C_{t0}|^\rho \exp \left( -\sum n_t E_t(x, \rho) \right), \quad \text{(14)}
\]

643/25/3-6
TABLE I

<table>
<thead>
<tr>
<th>Type index $l$</th>
<th>Description</th>
<th>$\lambda(j, k, k')$</th>
<th>$\mu_l$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$c$</td>
<td>Correct path only: $(y)$</td>
<td>$-\alpha m_{jk}$</td>
<td>0</td>
</tr>
<tr>
<td>$a$</td>
<td>Incorrect path only: $(y')$</td>
<td>$\alpha m_{jk'}$</td>
<td>1</td>
</tr>
<tr>
<td>$ca$</td>
<td>Correct and incorrect: $(y)$, $(y')$</td>
<td>$\alpha(m_{jk'} - m_{jk})$</td>
<td>1</td>
</tr>
</tbody>
</table>

where

$$
\exp \left( -E_c(\alpha, \rho) \right) \triangleq \sum_j \sum_k p_k p_{jk} \left( \frac{p_{jk}}{q_j} e^{-r} \right)^{-\alpha} ;
$$

$$
\exp \left( -E_a(\alpha, \rho) \right) \triangleq \sum_j q_j \left[ \sum_{k'} p_{k'} \left( \frac{p_{jk'}}{q_j} e^{-r} \right)^{\alpha} \right]^{\rho} ;
$$

$$
\exp \left( -E_{ca}(\alpha, \rho) \right) \triangleq \sum_j \sum_k p_k p_{jk} \left[ \sum_{k'} p_{k'} \left( \frac{p_{jk'}}{p_{jk}} \right)^{\rho} \right]^{\alpha} .
$$

Furthermore, we observe that since all correct path time slices are type $c$ or $ca$, and incorrect path slices are type $a$ or $ca$,

$$
nt = n_c + n_{ca} ;
$$

$$
n_{\tau} = n_a + n_{ca} = \sum n_{\mu_l} .
$$

Hence

$$
| C_{tr} | \leq M^\tau \\
= \exp n_{\tau} \\
= \exp \sum n_{\mu_l} ,
$$

and

$$
T_{tr}(\alpha, \rho) \leq \exp - \sum n_{t}[E_t(\alpha, \rho) - \mu_t \rho] .
$$

For the second bound, there are seven time-slice types, described in Table II. Here we let $\mu_l$ be the number of distinct incorrect paths in a type-$l$ time slice.
TABLE II

<table>
<thead>
<tr>
<th>Type index</th>
<th>Description</th>
<th>$\lambda_l(j, k, k_1, k_2)$</th>
<th>$\mu_1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$c$</td>
<td>(y)</td>
<td>$-2\alpha m_{jk}$</td>
<td>0</td>
</tr>
<tr>
<td>$a$</td>
<td>(y')</td>
<td>$\alpha m_{jk}$</td>
<td>1</td>
</tr>
<tr>
<td>$ca$</td>
<td>(y), (y')</td>
<td>$\alpha m_{jk} - 2\alpha m_{jk}$</td>
<td>1</td>
</tr>
<tr>
<td>$b$</td>
<td>(y', y')</td>
<td>$2\alpha m_{jk}$</td>
<td>1</td>
</tr>
<tr>
<td>$aa$</td>
<td>(y'), (y')</td>
<td>$\alpha m_{jk_1} + \alpha m_{jk_2}$</td>
<td>2</td>
</tr>
<tr>
<td>$cb$</td>
<td>(y), (y', y')</td>
<td>$2\alpha m_{jk} - 2\alpha m_{jk}$</td>
<td>1</td>
</tr>
<tr>
<td>$caa$</td>
<td>(y), (y'), (y')</td>
<td>$\alpha m_{jk_1} + \alpha m_{jk_2} - 2\alpha m_{jk}$</td>
<td>2</td>
</tr>
</tbody>
</table>

Then Lemma A3 gives, for $0 \leq \rho \leq 2$,

$$T_{\text{error}}(\alpha, \rho) \leq |C_{\text{error}}|^{\rho/2} \exp - \sum n_i E'_i(\alpha, \rho), \quad (19)$$

where

$$\exp -E'_c(\alpha, \rho) \triangleq \sum_j \sum_k P_k P_{jk} \left( \frac{P_{jk}}{q_j} e^{-r} \right)^{-\alpha \rho} \quad = \exp -E_c(\alpha, \rho);$$

$$\exp -E'_a(\alpha, \rho) \triangleq \sum_j \sum_k P_k P_{jk} \left[ \sum_{k'} P_{k'} \left( \frac{P_{jk'}}{q_j} e^{-r} \right)^{\alpha} \right]^{\rho/2};$$

$$\exp -E'_{ca}(\alpha, \rho) \triangleq \sum_j \sum_k P_k P_{jk} \left[ \sum_{k'} P_{k'} \left( \frac{P_{jk'}}{q_j} e^{-r} \right)^{2\alpha} \right]^{\rho/2};$$

$$\exp -E'_b(\alpha, \rho) \triangleq \sum_j \sum_k P_k P_{jk} \left[ \sum_{k'} P_{k'} \left( \frac{P_{jk'}}{q_j} e^{-r} \right)^{3} \right]^{\rho} = \exp -E_a(\alpha, \rho);$$

$$\exp -E'_{ab}(\alpha, \rho) \triangleq \sum_j \sum_k P_k P_{jk} \left[ \sum_{k'} \frac{P_{k'}}{P_{jk}} \right]^{2\alpha \rho/2};$$

$$\exp -E'_{ca}(\alpha, \rho) \triangleq \sum_j \sum_k P_k P_{jk} \left[ \sum_{k'} \frac{P_{k'}}{P_{jk}} \right]^{2\alpha \rho};$$

$$\exp -E'_{aa}(\alpha, \rho) \triangleq \sum_j \sum_k P_k P_{jk} \left[ \sum_{k'} \frac{P_{k'}}{P_{jk}} \right]^{\rho} = \exp -E_{ca}(\alpha, \rho).$$
Furthermore, by conservation of time-slice types,

\[ nt = n_e + n_{ea} + n_{eb} + n_{eaa}; \]
\[ nT_0 = n_e + n_{eb}; \]
\[ n(\tau_1 + \tau_2 - 2\tau_0) = n_a + n_{ea} + 2n_{aa} + 2n_{eaa}. \]

Consequently,

\[ nT_0 + n(\tau_1 - \tau_0) + n(\tau_2 - \tau_0) = \sum n_i\mu_i \]

and

\[ |C_{\tau_0\tau_1\tau_2}| \leq M_{\tau_0}M_{\tau_1-\tau_0}M_{\tau_2-\tau_0} = \exp \sum n_i\mu_i r, \]

so that

\[ T_{\tau_0\tau_1\tau_2}(\alpha, \rho) \leq \exp -\sum n_i \left[ E'_i(\alpha, \rho) - \mu_i \frac{\rho}{2} r \right]. \]

For Step 5, we apply Hölder's inequality to each of these quantities, getting

\[ \exp -E'_c(\alpha, \rho) \leq \exp \alpha \rho r - (1 - \alpha \rho)E_0 \left( \frac{\alpha \rho}{1 - \alpha \rho} \right) = \delta_c; \]
\[ \exp -E'_a(\alpha, \rho) \leq \exp \frac{\rho}{2} (1 - \alpha) r - \frac{\alpha \rho}{2} E_0 \left( \frac{1 - \alpha}{\alpha} \right) \leq \delta_a'; \]
\[ \exp -E'_e(\alpha, \rho) \leq \exp \frac{\rho}{2} (1 + \alpha) r - (1 - \alpha \rho)E_0 \left( \frac{\alpha}{1 - \alpha \rho} \right) = \delta_a'^2; \]
\[ \exp -E'_b(\alpha, \rho) \leq \exp \frac{\rho}{2} (1 - 2\alpha) r - \alpha \rho E_0 \left( \frac{1 - 2\alpha}{2\alpha} \right) = \delta_b; \]
\[ \exp -E'_{aa}(\alpha, \rho) - \rho r \leq \exp \rho (1 - \alpha) r - \alpha \rho E_0 \left( \frac{1 - \alpha}{\alpha} \right) = \delta_a'^2 = \delta_a; \]
\[ \exp -E'_{eb}(\alpha, \rho) - \rho r \leq \exp \frac{\rho}{2} (1 - \alpha \rho)E_0 \left( \frac{\alpha}{1 - \alpha \rho} \right) - \alpha \rho E_0 \left( \frac{1 - 2\alpha}{2\alpha} \right) = \delta_c\delta_b; \]
\[ \exp -E'_{eaa}(\alpha, \rho) - \rho r \leq \exp \rho (1 - \alpha \rho)E_0 \left( \frac{\alpha}{1 - \alpha \rho} \right) - \alpha \rho E_0 \left( \frac{1 - \alpha}{\alpha} \right) = \delta_c\delta_a'^2 = \delta_c\delta_a. \]
Here we have taken note of the remarkable fact that even though the correct path and incorrect paths are not statistically independent, the bounds we have obtained can be factored into products of the bounds $\delta_a$, $\delta_a'$, $\delta_b$, and $\delta_c$ we would have obtained had the paths been independent. This is why the result obtained in Appendix A in the special case of totally symmetric channels goes through to the general case.

Now let us examine the conditions under which all these terms are less than 1. We have

$$\delta_c < 1 \quad \text{if } r < \frac{1 - \alpha \rho}{\alpha \rho} E_0 \left( \frac{\alpha \rho}{1 - \alpha \rho} \right), \text{ or } \rho_r > \frac{\alpha \rho}{1 - \alpha \rho}, \text{ and } \alpha \rho > 0;$$

$$\delta_a' < 1 (\delta_a < 1) \quad \text{if } r < \frac{\alpha}{1 - \alpha} E_0 \left( \frac{1 - \alpha}{\alpha} \right), \text{ or } \rho_r > \frac{1 - \alpha}{\alpha}, \text{ and } 1 - \alpha > 0;$$

$$\delta_b < 1 \quad \text{if } r < \frac{2 \alpha}{1 - 2 \alpha} E_0 \left( \frac{1 - 2 \alpha}{2 \alpha} \right), \text{ or } \rho_r > \frac{1 - 2 \alpha}{2 \alpha}, \text{ and } 1 - 2 \alpha > 0. \quad (26)$$

If we choose $\rho < \rho_r$ and $\alpha = 1/(1 + \rho)$, then $\delta_a$ and $\delta_a'$ are less than 1, which is all we need to bound $T_{tr}(\alpha, \rho)$. For $\delta_b < 1$, since $(1 - \alpha)/\alpha > (1 - 2\alpha)/2\alpha$, we automatically have $\rho_r > (1 - 2\alpha)/2\alpha$, but we must also have $\alpha < 1/2$ or $\rho > 1$. Thus, using (16) and (21),

$$T_{tr}(\alpha, \rho) < \delta_c^{\alpha} \delta_a^{\alpha'} < 1, \quad \rho < \rho_r \text{ and } 0 \leq \rho \leq 1; \quad (27)$$

$$T_{tr(\alpha, \rho)} < \delta_c^{\alpha} \delta_b^{\alpha'} \delta_a^{\alpha'(1 + 2\alpha - 2\alpha')} < 1, \quad \rho < \rho_r \text{ and } 1 < \rho \leq 2.$$

Substituting back in (13), we have finally

$$\Pr(C_i' \geq N) \leq K N^{-\rho}, \quad (28)$$

where $\rho < \rho_r$ and $K$ is a constant given by

$$K \triangleq \begin{cases} \frac{1}{(1 - \delta_c)(1 - \delta_a)}, & 0 \leq \rho \leq 1; \\ \frac{1}{(1 - \delta_c)(1 - \delta_a)^2(1 - \delta_a)}, & 1 < \rho \leq 2. \end{cases} \quad (29)$$

This completes the proof of the main theorem for $0 \leq \rho_r \leq 2$. It should be clear that the method extends with additional bookkeeping to all rates:

**Theorem 2.** Let $\rho_r$ be defined by $r = E(\rho_r)|_{\rho_r}$. On any discrete memoryless
channel, for any ρ < ρr, the probability that the number of time-t computations \( C_t \) exceeds \( N \) is bounded by

\[
\Pr(C_t \geq N) \leq KN^{-e},
\]

where \( K \) is a finite constant independent of \( N \).

V. COMPUTATIONAL DISTRIBUTION: CONVERSE

A simple argument due to Jacobs and Berlekamp (1967) shows that no algorithm of the sequential decoding type can have a computational distribution better than the Pareto distribution of Theorem 2.

The argument resembles the error probability lower bounds in relying on block code results applied to a truncated convolutional code of some critical length. We use the Shannon, Gallager and Berlekamp (1967) result that there are asymptotically no block codes of length \( N \) and with \( M \) code words, such that with list-of-\( L \) decoding the error probability is less than \( \exp -NE_{sp}(R_{eff}) \), where \( R_{eff} \) is the effective rate

\[
R_{eff} = \frac{\ln M - \ln L}{N},
\]

and \( E_{sp}(R_{eff}) \) is the sphere-packing bound

\[
E_{sp}(R_{eff}) = E_0(\rho_{eff}) - \rho_{eff} R_{eff},
\]

where \( \rho_{eff} \) satisfies \( R_{eff} = E_0'(\rho_{eff}) \). [It is implicit in all this that we use the best \( p \) in \( E_0(p) \).]

Now, for a trellis code of rate \( r \), consider the set \( C_{crit} \) of all incorrect paths in \( S_0 \) truncated to length \( N_{crit} \), where

\[
N_{crit} = \frac{\rho_r \ln L}{E_0(\rho_r) - \rho_r E_0'(\rho_r)}
\]

and \( \rho_r \) satisfies \( r = E_0(\rho_r)/\rho_r \). Together with the correct path truncated to the same length, these paths form a block code of length \( N_{crit} \), and with a number of words asymptotically equal to \( rN_{crit} \). The effective rate of this code is thus

\[
R_{eff} = r - \frac{\ln L}{N_{crit}} = r - \frac{E_0(\rho_r)}{\rho_r} + E_0'(\rho_r) = E_0'(\rho_r).
\]
Hence $\rho_{\text{eff}} = \rho_r$. The probability that the correct path metric is less than $L$ different incorrect path metrics at depth $N_{\text{crit}}$ is the probability of list-of-$L$ decoding error, which asymptotically must be at least as great as

$$\Pr_L(\mathcal{E}) \geq \exp \{-N_{\text{crit}} \left[ E_0(\rho_r) - \rho_r \frac{\ln L}{N_{\text{crit}}} \right] \} = L^{-\rho_r}. \quad (35)$$

In other words, the probability that there are $L$ or more incorrect path metrics above the correct path metric at depth $N_{\text{crit}}$ is at least (asymptotically) $L^{-\rho_r}$ for any trellis code whatsoever of rate $r$.

The argument is completed by observing that any sequential decoding algorithm as defined in Definition 1 must examine the nodes at depth $N_{\text{crit}}$ in some order; that on the average an algorithm will examine the minimum number of paths at depth $N_{\text{crit}}$ if it examines them in order of decreasing metric, since those with greater metric have greater likelihood of being correct; and therefore that the probability of having to examine at least $L$ paths is at least $L^{-\rho_r}$, asymptotically. For the rigorous argument, see Jacobs and Berlekamp (1967).

We note that the critical length $N_{\text{crit}}$ as defined here is equal to the critical length in the error probability derivations of Part II if $0 \leq \rho_r \leq 1$ and we choose a constraint length $v_L$ such that

$$M^{v_L} = L. \quad (36)$$

Furthermore, if a sequential decoding algorithm can perform a maximum number of computations $\lambda$ in one incorrect subset $S_i$, then the probability of decoding (buffer overflow) is asymptotically

$$\Pr(\text{failure}) \sim \lambda^{-\rho_r} = M^{-v_r \rho_r}, \quad (37)$$

which is the error probability for a maximum-likelihood decoder with a code of constraint length $v = \log_M \lambda$. In this sense, regardless of the true constraint length $v$, the “equivalent” constraint length of a sequential decoder is $v$, when $v = v$, and all the “action” occurs at the critical length

$$N_{\text{crit}} = \frac{nv_r E_0(\rho_r)}{E_0(\rho_r) - \rho_r E_0'(\rho_r)}. \quad (38)$$

If the code had constraint length $v$, the fraction of the total paths of length
$N_{\text{crit}}$ in $S_0$ remerging at depth $N_{\text{crit}}$ would be $M^{-\rho_0}$; the probability that one of these remerging paths would be more likely than the correct path is the same as the probability that $M^{\rho_0}$ of all the paths of length $N_{\text{crit}}$ are more likely than the correct path, for $0 \leq \rho \leq 1$ ($R_{\text{comp}} \leq r \leq C$).

VI. ERROR PROBABILITY

We now show that in the high-rate range, $R_{\text{comp}} \leq r \leq C$ ($0 \leq \rho \leq 1$), the error probability of Algorithm A is asymptotically identical to that of maximum-likelihood decoding.

**Theorem 3.** Let $\rho_0$ be defined by $r = E_0(\rho_0)/\rho_0$. On any discrete memoryless channel, for any $\rho < \rho_0$ satisfying $0 \leq \rho \leq 1$, the probability of error per unit time with Algorithm A satisfies

$$\Pr(\mathcal{E}) \leq K \exp -n\rho r,$$

where $K$ is a constant independent of $n$.

**Proof.** We actually bound the probability of proto-error. A proto-error occurs only if

$$I(y_{\tau'}) \geq I(y_{\tau}) - \gamma_{\tau}$$

where $y_{\tau'}$ is some possible error event in the incorrect subset $S_{\tau}$. Again we let $\tau = 0$ and $I(y_{\tau}) = 0$. Define $C_{\tau}$ as the set of possible error events in $S_0$ that remerge with the correct path at time $\tau$; then

$$\Pr(\mathcal{E}_0) \leq \sum_{\tau=0}^{\infty} \left[ \sum_{y_{\tau'} \in C_{\tau}} \exp \alpha(I(y_{\tau'}) + \gamma_0) \right]^\rho, \quad \alpha \geq 0, \quad 0 \leq \rho \leq 1.$$  \hspace{1cm} (41)

With substitution of (10) we have

$$\Pr(\mathcal{E}_0) \leq \sum_{t=0}^{\infty} \sum_{\tau=\rho+1}^{\infty} \left[ \sum_{y_{\tau'} \in C_{\tau}} \exp \alpha[I(y_{\tau'}) - I(y_{\tau})] \right]^{\rho}, \quad \alpha \geq 0, \quad 0 \leq \rho \leq 1;$$

$$= \sum_{t=0}^{\infty} \sum_{\tau=\rho+1}^{\infty} T_{t\tau}(x, \rho).$$  \hspace{1cm} (42)

The term $T_{t\tau}(x, \rho)$ is the same as $T_{t\tau}(x, \rho)$ as defined in (13), except that the configuration $C_{\tau}$ has $|C_{\tau}| \leq M^{-\rho}$, whereas $|C_{\tau}| \leq M^\tau$. Hence the proof
goes through as in the proof of Theorem 2 (for \(0 \leq \rho \leq 1\)), except for an additional term \(\exp -nvpr\). We get

\[
T_v'(\alpha, \rho) \leq \delta_v^{nt} \delta_o^{nv} \exp -nvpr, \quad \rho < \rho_r \quad \text{and} \quad 0 \leq \rho \leq 1, \tag{43}
\]

where \(\delta_o\) and \(\delta_a\) < 1. Consequently, substituting (43) in (42),

\[
Pr(\mathbf{\theta}_0) \leq K \exp -nvpr, \tag{44}
\]

where

\[
K \triangleq \frac{\delta_a^{n(\rho + 1)}}{(1 - \delta_e^n)(1 - \delta_a^n)}. \tag{45}
\]

For \(0 \leq \rho_r \leq 1\) \([C \geq r \geq R_{\text{comp}} = E_0(1)\]), we may let \(\rho\) approach \(\rho_r\), so that \(\rho_r\) approaches \(E_0(\rho_r)\), and we have asymptotically the same result as in Theorem 1 of Part II for maximum-likelihood decoding. In this range Theorem 3 of Part II shows the bound to be asymptotically optimum and hence tight.

For \(\rho_r > 1\), or \(r < R_{\text{comp}}\), Theorem 3 gives only \(Pr(\mathbf{\theta}) \leq K \exp -nr\), whereas with maximum-likelihood decoding \(Pr(\mathbf{\theta}) \leq K \exp -nR_{\text{comp}}\). It can be shown that any possible incorrect path of length \(v + 1\) has probability \(\sim \exp -nr\) of having metric greater than zero at the remerging point (if the metric is zero at the diverging point); hence no tighter bound than Theorem 3 is possible in general. The error probability can be shown to be asymptotically equal to the optimum \(\exp -nR_{\text{comp}}\) if we replace \(r\) by \(R_{\text{comp}}\) in the metric increments,

\[
m_{jk} = \ln \frac{p_{jk}}{q_j} - R_{\text{comp}}, \tag{46}
\]

at a sacrifice, however, of Pareto exponent in the computational distribution. Since decoder complexity is far more a function of computational distribution than of constraint length \(v\), it is generally preferable to leave the metric increments alone and make \(v\) sufficiently large that error probability is negligible. In any case, the suboptimality is small at the rates near \(R_{\text{comp}}\) where sequential decoders are usually operated.

VII. EXTENSION OF RESULTS TO OTHER ALGORITHMS

We now outline proofs that the same asymptotic computational distribution and error probability results apply to the other algorithms cited above, as well as to the popular Fano algorithm. [For additional discussion of algorithms see Geist (1973).]
In Algorithm Z, we prune the stack by eliminating paths whose metrics are more than $\beta$ below the top path. This can eliminate the correct path only if the correct path dips by at least $\beta$, which is shown in Appendix A (Theorem 4) to be an event of probability asymptotically

$$\exp - \frac{\rho_r}{1 + \rho_r} \beta.$$

(47)

If we choose $\beta$ large enough so that

$$\exp \frac{1}{1 + \rho_r} \beta > \max\{M_r, \lambda\},$$

(48)

then neither error probability nor computational distribution is significantly affected by this refinement.

In Algorithm $Z_J$, we forget about eliminations due to merging. From the fact that the error probability bound of Theorem 3 is actually a bound on the proto-error event, and that a remerging node can never even be examined unless a proto-error occurs, it is clear that error probability is asymptotically indifferent to whatever we might do when paths merge: discard the worst, discard the one examined later, or let the basic algorithm choose a winner. There may, however, be additional computations if we forget about merging. First, within an incorrect subset paths may remerge. This introduces more complicated configurations than occur without merging (with effectively infinite constraint length), but it appears that more detailed configuration-counting would yield asymptotically unchanged results in the incorrect subset. Second, incorrect paths may merge with the correct path; if one is not eliminated, then the other is carried along as “excess baggage.” To see that this effect is negligible, suppose that there are $m$ paths all merged with the correct path being carried along as excess baggage at some time, each with its own metric. If the correct path dips at all subsequent to this time, then the best of these paths will eventually “win,” and in the worst case (all metrics equal) we will search $m$ identical incorrect subsets down to the dip, thus multiplying our computation by no more than $m$. If the correct path does not dip subsequently, then the first of these paths to be examined “wins,” and no excess computation ensues. Since the probability of even one extra path is bounded by the proto-error bound, which decreases exponentially with $\nu$, it is plausible that the excess computations are negligible. (A rigorous proof would require a list-of-$L$ proto-error probability bound for arbitrarily large $L$.)

In Algorithm $J$, we quantize the metrics into bins of width $\Delta$. It is easy to see that in this case a node $y'$ may be examined only if $\Gamma(y') \geq \Gamma(y_t) - \gamma_t - \Delta$. 

The computational distribution and error probability bounds (Theorems 2 and 3) carry through as before with an additional factor,

\[ \exp \alpha p \Delta \to \exp \frac{\rho_r}{1 + \rho_r} \Delta, \]  \hspace{1cm} (49)

multiplying the constant \( K \). Evidently for modest \( \Delta \) this is insignificant.

Finally, let us consider the Fano (1963) algorithm. Without going into details, which may be found in Gallager (1968), we may describe the Fano algorithm as follows. As in Algorithm J, a series of thresholds spaced by \( \Delta \) is set up. With each threshold is associated a search; the search begins when the correct path first crosses the threshold for good ("breakout") and ends when it first crosses the next higher threshold for good. Of course the decoder does not know which is the correct path and can never be sure of having crossed a threshold "for good," but the algorithm is set up to hypothesize recursively tentative breakouts whenever paths first cross a new threshold, and to resume a previous search whenever all paths stemming from a tentative breakout node fail to stay above the corresponding threshold "for good," so that the tentative breakout is not a true breakout. The important properties of the Fano algorithm are:

1. Searches are made by backward and forward steps through the code trellis along contiguous branches. As a practical matter this means only the current node history need be stored, not an entire stack of histories. This is why the Fano algorithm is universally preferred in hardware sequential decoders, although stack algorithms can be more efficient in software on general-purpose computers with large memories (Geist, 1971).

2. An incorrect path is searched once for each threshold lying between its metric and the next threshold below the minimum on the correct path after the diverging point. A correct path dip of \( \gamma \) therefore leads to as many as \( \gamma / \Delta \) extra searches on each node. The increase in the total number of computations is therefore bounded by a factor of \( \beta / \Delta \), where \( \beta \) is the maximum allowable correct path dip. Thus although the number of computations is greater with the Fano algorithm than with stack algorithms, the distribution of computation remains Pareto with the same exponent \( \rho_r \).

VIII. COMPLEXITY

We return at last to the gut issue of complexity. We shall show that maximum-likelihood decoding and sequential decoding have the same
asymptotic relation between complexity and performance, but that that of sequential decoding occurs in much more palatable form.

In Part II we defined the complexity $G$ of a maximum-likelihood decoder as $M^r$, the total number of states, since both the number of computations per unit time and the total storage are proportional to $M^r$. With this definition, the error probability as a function of $G$ is

$$\Pr(\delta) \simeq M^{-\rho r} \simeq G^{-\rho r}, \quad 0 \leq \rho_r \leq 1,$$

in the high-rate range, $R_{\text{comp}} \leq r \leq C$.

In Appendix B we show that with Viterbi's (1967) "semisequential" maximum-likelihood decoding modified to use a list-of-$L$ decoder, the above result is extended to all rates, using a definition of complexity in which $G$ is the maximum number of computations per unit time or histories needing to be stored. The average may be much less (for $\rho_r > 1$), rather than constant as in ordinary maximum-likelihood decoding.

A reasonable definition of complexity for sequential decoding is $G = \lambda$, the maximum number of computations which can be performed in one incorrect subset. This measure ignores the storage requirements of the stack algorithms, but these can be avoided if desired by use of the Fano algorithm. We have seen that the probability of decoding failure (buffer overflow) is then given by

$$\Pr(\text{failure}) \simeq G^{-\rho r}$$

for any rate, just as above.

However, here $G$ does not represent a fixed amount of storage that must be available, but rather a capability for a certain maximum number of computations. When $\rho_r > 1$, the Pareto distribution of computations per unit time has a mean value. If the number of computations $\mu$ that the decoder can perform per unit time is greater than this mean value, then on the average the decoder can keep up with the data, although buffering may be necessary. If $\rho_r \leq 1$, however, the mean number of computations required is infinite. This is the reason that, although decoding at finite error probabilities may be carried on when $\rho_r \leq 1$, the computational cutoff rate $R_{\text{comp}}$ corresponding to $\rho_r = 1$ is considered the nominal upper limit for sequential decoders.

In real-time situations, where a requirement for putting out data with a fixed buffering delay of $B$ time units exists, the maximum number of computations in any one incorrect subset is clearly limited to $\mu B$, the total number of computations available in one delay interval. For $\rho_r > 1$, or $R < R_{\text{comp}}$, and for a speed factor $\mu$ somewhat larger than the mean number of computa-
tions per unit time, this maximum number is nearly always available (Jordan, 1966), so that effectively
\[
Pr(\delta) \sim (\mu B)^{-\tau r}.
\] (52)

Experimental evidence shows that the constant of proportionality is usually in the range 1–10 for the Fano algorithm. Currently $\mu B$ products of $10^8$ to $10^9$ for delays of less than 1 sec are achievable in hardware (Forney and Bower, 1971), so that good performance is obtainable right up to $R_{\text{comp}}(\rho_r = 1)$ and even somewhat beyond. On the other hand, hardware complexity limits maximum likelihood decoders to $G \sim 10^2$ to $10^3$ states. It is difficult to imagine that the progress of technology will change these ratios radically. We would therefore expect that sequential decoders will continue to be preferred over maximum-likelihood decoders whenever maximum performance is demanded, as long as the data rate is low enough that sequential decoders can keep up on the average. Maximum-likelihood decoders may be preferred, however, when the performance demanded is more modest, the data rate extremely high, or the allowable decoding delay small.

It is interesting to note that for digital communication via space satellites, maximum-likelihood (Viterbi) decoders are presently chosen over sequential decoders. In large part this is due to the common requirement to operate at megabit data rates, where Viterbi decoders can utilize "soft decisions" but sequential decoders cannot and thus lose their performance advantage. Even at more moderate data rates Viterbi decoders are often preferred because of their greater simplicity and greater robustness to various suboptimal channel and receiver conditions. However, in deep space telemetry, where speeds are modest and every last dB is vital, as well as in applications requiring extremely low data rates, sequential decoders still seem to be preferred.

The Pareto distribution of computation and complexity, which the Jacobs-Berlekamp result shows to be inevitable with sequential decoding algorithms, is disagreeable for two reasons. First, in theory, one would like a scheme whose probability of error would decrease exponentially rather than algebraically with the complexity parameter. Epstein (1958) showed that this was possible with convolutional codes on a special channel, the erasure channel. That it is possible in general at all rates less than capacity is demonstrated by block concatenated codes (Forney, 1966), whose error probability decreases exponentially with block length while decoding complexity increases algebraically. (As yet, however, there are no memoryless channel applications known to the author where concatenated codes outperform sequential decoding schemes of comparable complexity.) Second, one would like to remove the limitation to rates below $R_{\text{comp}}$. The most promising attempt in
this direction is that of Jelinek and Cocke (1971), whose scheme elaborates on Falconer's (1969) idea of parallel convolutionally encoded streams with algebraic cross-constraints. Progress on solving either of these problems would be both theoretically and practically significant.

APPENDIX A: MARTINGALE APPROACH TO COMPUTATIONAL BOUNDS

In this appendix we shall quickly, and without regard for rigor, develop some interesting partial results having to do with the computational distribution of an Algorithm A sequential decoder. As many others have done, we observe that the probability that the correct path dips by \( \beta \) decreases exponentially with \( \beta \), while the number of nodes in the correct subset above the dip increases exponentially with \( \beta \), in such a way as to suggest the Pareto distribution \( \Pr(C_t \geq N) \sim N^{-\alpha_r} \). The argument can be made rigorous only for "totally symmetric" channels, where the statistics of the correct path and the incorrect subset are actually independent; Section IV derives the general rigorously for all channels (for \( 0 < \rho \leq 2 \)).

Again, we shall assume a random \((M, \nu, n)\) trellis code characterized by a channel input \( p \), where \( p_k = \Pr(y_i = k) \), and a discrete memoryless channel characterized by transition probabilities \( \{p_{jk}\} \), where \( p_{jk} = \Pr(j | k) \). We have previously (Forney, 1968) defined a "totally symmetric" channel as one for which \( \sum p_k p_{jk} \) is independent of \( j \) for any \( \alpha \)—e.g., the binary symmetric channel with \( p = \{1/2, 1/2\} \).

The metric of the correct path \( \Gamma(y_i) \) then performs a random walk, with the probability of metric increment \( m_{jk} \) given by \( \Pr(k) \Pr(j | k) = p_k p_{jk} \).

Martingale methods are useful for quick proofs of certain random walk properties. A sequence \( \{x_t\} \) is a martingale if \( \mathbb{E}[x_{t+1} | H_t] = x_t \), where by \( H_t \) we mean the entire history of the sequence to time \( t \). By induction \( \mathbb{E}[x_\tau | H_t] = x_t \) for any \( \tau \geq t \). The sequence is a lower semi-martingale if \( \mathbb{E}[x_{\tau+1} | H_t] \leq x_t \), whence \( \mathbb{E}[x_\tau | H_t] \leq x_t \), \( t \leq \tau \).

To estimate the probability that, starting at \( \Gamma(y_i) \), the correct path ever dips below \( \Gamma(y_i) - \beta \), so that \( \gamma_t \geq \beta \), we define \( \Gamma'(y_i) \) as a random walk that follows \( \Gamma(y_i) \) unless \( \Gamma'(y_i) \) drops below \( \Gamma(y_i) - \beta \), where it "sticks":

\[
\Gamma'(y_i) = \begin{cases} 
\Gamma(y_i), & \Gamma'(y_{\tau-1}) > \Gamma(y_i) - \beta; \\
\Gamma'(y_{\tau-1}), & \Gamma'(y_{\tau-1}) \leq \Gamma(y_i) - \beta.
\end{cases}
\]  

When the general trend of \( \Gamma(y_i) \) is upward, \( \Gamma'(y_i) \) either goes to infinity as \( \tau \to \infty \), or else sticks at a value below \( \Gamma(y_i) - \beta \). Consider now the quantity

\[
T_\tau(\alpha) = \exp -\alpha \Gamma'(y_i), \quad \alpha \geq 0.
\]
Lemma 2. Let $\rho_r$ be defined by $r = E_0(\rho_r)/\rho_r$. On totally symmetric channels, $T_r(\alpha)$ is a martingale if $\alpha = \rho_r/(1 + \rho_r)$. On any discrete memoryless channel, $T_r(\alpha)$ is a lower semi-martingale if $\alpha \leq \rho_r/(1 + \rho_r).

Proof. We have

$$E[T_{r+1}(\alpha) | H_r] = \begin{cases} \exp -\alpha [I''(y_r) + \lambda_{r+1}], & I''(y_r) > I'(y_r) - \beta; \\ \exp -\alpha I''(y_r), & I''(y_r) \leq I'(y_r) - \beta; \end{cases}$$

(55)

where

$$\lambda_{r+1} = \sum_{i=r+1}^{(r+1)n} m_i$$

(56)

is the sum of the metric increments $m_i$, $\tau n < i \leq (\tau + 1)n$, on the correct path branch $\tau + 1$. Since these are independent of $H_r$ and of each other, we have

$$\exp -\alpha [I''(y_r) + \lambda_{r+1}] = \exp -\alpha I''(y_r) [\exp -\alpha m_i]^n$$

(57)

since all $n$ averages are identical. Hence $T_r(\alpha)$ is a martingale if the bracketed quantity $\exp -\alpha m_i$ equals 1, and a lower semi-martingale if it is overbounded by 1. By Hölder's inequality

$$\exp -\alpha m_i = \sum_j \sum_k p_k p_{jk} \exp -\alpha m_{jk}$$

$$\leq e^{\alpha r} \left[ \sum_j q_j \right]^n \left[ \sum_k \left( \sum_j p_k p_{jk}^{1-\alpha} \right)^{1/(1-\alpha)} \right]^{1-\alpha}, \quad 0 \leq \alpha \leq 1,$$

$$= \exp \alpha r - (1 - \alpha) E_0 \left( \frac{\alpha}{1 - \alpha} \right),$$

$$= \delta,$$  

(58)

where we have used $\sum q_j = 1$, introduced Gallager's function $E_0(\rho)$, and denoted the last expression by $\delta$. We note that equality holds if and only if $\sum p_k p_{jk}^{1-\alpha}$ and $q_j = \sum p_k p_{jk}$ are independent of $j$—i.e., on totally symmetric channels. Now $\delta = 1$ if and only if

$$r = \frac{1 - \alpha}{\alpha} E_0 \left( \frac{\alpha}{1 - \alpha} \right),$$

(59)
or
\[ \rho_r = \frac{\alpha}{1 - \alpha}, \]
(60)

or
\[ \alpha = \frac{\rho_r}{1 + \rho_r}. \]
(61)

Similarly, from the monotonicity of \( E_0(\rho)/\rho \) with \( \rho, \delta \leq 1 \) if \( \alpha \leq \rho_r/(1 + \rho_r) \).

Q.E.D.

Consequently \( T_\infty(\alpha) \leq \exp -\alpha \Gamma(y_t) \) for \( \tau \geq t, \alpha \leq \rho_r/(1 + \rho_r) \). As \( \tau \to \infty, T_\infty(\alpha) \) approaches zero if the correct path [and hence \( \Gamma''(y_t) \)] never dips below the threshold, or sticks at some value \( \geq \exp -\alpha[\Gamma(y_t) - \beta] \) if \( \Gamma(y_t) \) ever goes below \( \Gamma(y_t) - \beta \). We can argue as follows:

\[
\lim_{\tau \to \infty} T_\infty(\alpha) \geq \Pr(\gamma_t \geq \beta) \mathbb{E}[\exp -\alpha \Gamma'(y_t) \mid \gamma_t \geq \beta] \\
\geq \Pr(\gamma_t \geq \beta) \exp -\alpha[\Gamma(y_t) - \beta].
\]
(62)

Hence
\[
\Pr(\gamma_t \geq \beta) \leq T_\infty(\alpha) \exp -\alpha[\Gamma(y_t) - \beta] \\
\leq \exp -\alpha \beta, \quad \alpha \leq \rho_r/(1 + \rho_r).
\]
(63)

The best bound is obtained by using \( \alpha = \rho_r/(1 + \rho_r) \).

**Theorem 4.** The probability that the correct path dip exceeds \( \beta \) is bounded by
\[
\Pr(\gamma_t \geq \beta) \leq \exp -\frac{\rho_r}{1 + \rho_r} \beta.
\]
(64)

The martingale argument can be used to show that this bound is asymptotically tight.

Now let us examine the behavior of the incorrect subset. The incorrect subset \( S_t \) forms what has been called a "branching random walk," in that at each time \( \tau \) each node splits into \( M \) branches, each of which then executes independent branching random walks. The probability of the metric increment \( m_{nk} \) on a branch unmerged with the correct path is \( p_k q_i = p_k \sum_k p_k p_{ik} \).

As in Section IV, to avoid considering merging, we note that in Algorithm A, \( C_t \leq C_t' \), where \( C_t' \) is the number of nodes \( y' \in S_t' \), and \( S_t' \) is the time-\( t \) incorrect subset of an \((M, \infty, n)\) code. Henceforth \( S_t' \) is to be regarded in this light.

Again the martingale approach is a quick path to the desired results. Consider the branching random walk in which any path in the incorrect
subset that drops below $\Gamma(y_t) = \gamma_t$ simply disappears. Let the set of paths $y_t'$ which survive at time $\tau$ be $S_t \subset S_t'$. Consider

$$T_\tau(\alpha) = \sum_{y_t' \in S_t'} \exp \alpha \Gamma(y_t');$$

$$T_\tau'(\alpha) = \sum_{y_t' \in S_t} \exp \alpha \Gamma(y_t'),$$

(65)

where the second sum is only over paths which have never dropped below the threshold.

**Lemma 3.** Let $\rho_r$ be defined by $r = E_0(\rho_r)/\rho_r$. On totally symmetric channels, $T_\tau(\alpha)$ is a martingale for $\alpha = 1/(1 + \rho_r)$. On any discrete memoryless channel, $T_\tau'(\alpha)$ is a lower semi-martingale for $\alpha > 1/(1 + \rho_r)$. In fact, for $\alpha > 1/(1 + \rho_r)$, $T_\tau'(\alpha) \leq \delta^{n(\tau-\tau_0)} T_\tau(\alpha)$ for some $\delta < 1$.

**Proof.** For the definition of $T_\tau(\alpha)$ given by (54), we have

$$\mathbb{E}[T_{\tau+1}(\alpha) \mid H_\tau] = \sum_{\text{all } y_t' \in S_t} \exp \alpha \Gamma(y_t') \sum_{\text{extensions of } y_t'} \exp \alpha \sum_{i=\tau+1}^{(\tau+1)n} m_i'$$

$$= \sum_{\text{all } y_t' \in S_t} \exp \alpha \Gamma(y_t') M \left[ \sum_{j} \sum_{k'} P_{k'} q_j \exp \alpha m_{j'} \right]$$

$$= T_\tau(\alpha) \left[ e^\alpha \sum_{j} \sum_{k'} P_{k'} q_j \left( \frac{P_{j'}}{q_j} e^{-\tau} \right)^{\alpha} \right]^n,$$

(66)

where we have recognized that all $m_i'$, $\tau n < i \leq (\tau + 1)n$, are independent of each other and of $H_\tau$, recognized that to each $y_t' \in S_t$ there correspond $M$ identical terms, one for each branch from $y_t'$, and substituted $M = \exp n\tau$. We get a similar expression for $T_{\tau+1}(\alpha)$, except that since fewer than $M$ extensions may stay above the threshold we get an inequality. (We would get equality if paths were made to "stick" below the threshold as in Lemma 2 rather than disappear.) The bracketed term is again bounded by Hölder's inequality:

$$e^{(1-\alpha)\tau} \sum_j q_j^{1-\alpha} \sum_{k'} P_{k'} \sum_{j'} q_j' \leq e^{(1-\alpha)\tau} \left[ \sum_j q_j \right]^{1-\alpha} \left[ \sum_k P_k \sum_{j'} q_j' \right]^{\alpha},$$

$$= \exp(1 - \alpha) \tau - \alpha E_0 \left( \frac{1 - \alpha}{\alpha} \right)$$

$$= \delta,$$

(67)
equality again holding on and only on totally symmetric channels. We have
\[ \delta = 1 \] if
\[ r = \frac{E_0(\rho_r)}{\rho_r} = \frac{\alpha}{1 - \alpha} E_0 \left( \frac{1 - \alpha}{\alpha} \right), \tag{68} \]
or \[ \rho_r = \frac{(1 - \alpha)}{\alpha}, \] or \[ \alpha = \frac{1}{(1 + \rho_r)}; \] similarly \[ \delta \leq 1 \] if \[ \alpha \geq \frac{1}{(1 + \rho_r)}. \]
In sum, on totally symmetric channels,
\[ E[T_{\tau+1}(\alpha)|H_r] = T_{\tau}(\alpha) \delta^n = T_{\tau}(\alpha), \quad \alpha = 1/(1 + \rho_r); \tag{69} \]
on all channels
\[ E[T'_{\tau+1}(\alpha)|H_r] \leq T'_{\tau}(\alpha) \delta^n \leq T'_{\tau}(\alpha), \quad \alpha \geq 1/(1 + \rho_r). \tag{70} \]
Finally, from the strict monotonicity of \[ E_0(p)/p, \delta < 1 \] if \[ \alpha > \frac{1}{(1 + \rho_r)}, \] and
\[ \frac{T_{\tau}(\alpha)}{\delta^{n(\tau-t)}} T'_{\tau}(\alpha) = \delta^{n(\tau-t)} \exp \alpha \Gamma(y_t) \tag{71} \]
decreases exponentially with \( \tau \) for \( \alpha > 1/(1 + \rho_r) \).
Q.E.D.

Let \( N_\tau \) be the number of nodes above \( \Gamma(y_t) - \gamma_t \) at time \( \tau \), and \( \bar{N}_\tau \) the corresponding average. Since all such nodes are in \( S_t \), and have \( \exp \alpha \Gamma(y_t) \geq \exp [\Gamma(y_t) - \gamma_t] \), we have
\[ T'_{\tau}(\alpha) \geq N_\tau \exp \alpha [\Gamma(y_t) - \gamma_t]; \]
\[ \bar{T}'_{\tau}(\alpha) \geq \bar{N}_\tau \exp \alpha [\Gamma(y_t) - \gamma_t]; \tag{72} \]
\[ \bar{N}_\tau \leq \bar{T}'_{\tau}(\alpha) \exp -\alpha [\Gamma(y_t) - \gamma_t] \leq \delta^{n(\tau-t)} \exp \alpha \gamma_t, \]
where \( \delta < 1 \) for \( \alpha > 1/(1 + \rho_r) \). Hence the mean of \( C_t' \) satisfies
\[ \bar{C}_t' = \bar{N} = \sum_{\tau=t}^{\infty} \bar{N}_\tau \leq \frac{\exp \alpha \gamma_t}{1 - \delta^n}, \tag{73} \]
which is what we want.

**Theorem 5.** For any \( \alpha > 1/(1 + \rho_r) \), for a given dip \( \gamma_t \), the average number of nodes in \( S_t \) that exceed \( \Gamma(y_t) - \gamma_t \) is bounded by
\[ \bar{C}_t \leq K \exp \alpha \gamma_t, \tag{74} \]
where \( K \) is a constant independent of \( \gamma_t \).
The fact that $C_t$ is bounded means that with probability 1 the whole incorrect subset dies out, or $S_t$, disappears as $\tau \to \infty$. This justifies our assertion that even the best path in the incorrect subset tends downward.

Theorems 4 and 5 show that while the probability of a dip greater than $\beta$ decreases exponentially with $\beta$, the mean number of nodes to be examined increases exponentially with $\beta$; hence it is tempting to conclude from (64) and (74) that

$$\Pr(C_t \geq N) \approx \Pr[\gamma_t \geq (1 + \rho_t) \ln N]$$
$$\approx \exp -\rho_r \ln N$$
$$= N^{-\rho_r}, \quad (75)$$

which is the Pareto distribution we seek. Such an argument turns out to have heuristic merit, even though it is fallacious in two respects: first, the correct path and incorrect subset are not independent except on totally symmetric channels; second, we have shown only $E[C_t | \gamma_t] \approx \exp \gamma_t/(1 + \rho_r)$, not that there are definitely $\exp \gamma_t/(1 + \rho_r)$ nodes to be examined when the dip is $\gamma_t$.

We need to go to Section IV to treat the general channel, but the latter deficiency can be corrected at high rates on the totally symmetric channel by use of the first-moment inequality

$$\Pr(x \geq x_0) \leq \bar{x}/x_0. \quad (76)$$

**Theorem 6.** Let $\rho_r$ be defined by $E_0(\rho_r)/\rho_r = r$, and let $0 \leq \rho_r \leq 1$, or $R_{\text{comp}} \leq r \leq C$. On totally symmetric channels, for any $\rho < \rho_r$,

$$\Pr(C_t \geq N) \leq K'N^{-\rho}, \quad (77)$$

where $K'$ is a constant independent of $N$.

**Proof.** For any $\beta_0$ and $k$, with $\Delta = \beta_0/k$, we have

\begin{align*}
\Pr(C_t \geq N) & \leq \Pr(\gamma_t \geq \beta_0) + \sum_{i=0}^{k-1} \Pr[\beta_0 - i\Delta > \gamma_t \geq \beta_0 - (i + 1)\Delta] \\
& \leq \Pr(\gamma_t \geq \beta_0) + \sum_{i=0}^{\infty} \Pr[\gamma_t \geq \beta_0 - (i + 1)\Delta] \\
& = \Pr[C_t \geq N | \gamma_t = \beta_0 - i\Delta]. \quad (78)
\end{align*}
We use the first-moment inequality
\[ \Pr(C_t \geq N | \gamma_t) \leq \frac{E[C_t | \gamma_t]}{N} \]
and Theorems 4 and 5, with the substitution \( \beta_0 = (1 + \rho_r) \ln N \), to get
\[ \Pr(C_t \geq N) \leq N^{-\alpha r} + \sum_{i=0}^{\infty} \left[ N^{-\alpha r} \exp \frac{\rho_r}{1 + \rho_r} (i + 1) \Delta \right] \left[ KN^{-\alpha (1+\rho_r)} \exp -\alpha i \Delta \right], \]
where \( \alpha > 1/(1 + \rho_r) \). Let \( \alpha(1 + \rho_r) = 1 + \epsilon \) for some arbitrarily small \( \epsilon > 0 \). Then
\[ \Pr(C_t \geq N) \leq N^{-\alpha r} + K \exp \frac{\rho_r}{1 + \rho_r} \Delta N^{-\alpha r + \epsilon} \sum_{i=0}^{\infty} \exp - \left( \frac{1 + \epsilon - \rho_r}{1 + \rho_r} \right) i \Delta \]
\[ \leq K' N^{-\alpha r + \epsilon}, \]
where the latter sum converges for any \( \epsilon > 0 \) and \( \rho_r \leq 1 \). Q.E.D.

Similarly we can get the Pareto distribution for \( 1 < \rho_r \leq 2 \) by showing that \( C_t^2 \) is bounded and asymptotically equal to \( C_t^2 \) for \( r < R_{\text{comp}} \), for \( 2 < \rho_r \leq 3 \) by showing the third moment is bounded, and so forth. The tightest result, which applies to all channels and seems to be asymptotically correct, is that for any \( \gamma_t \) the \((1 + \rho)\)th moment of the random variable \( C_t \) is bounded for \( \rho < \rho_r \). This can be deduced from the following lemma.

**Lemma 4.** Let \( \rho_r \) be defined by \( r = E_0(\rho_r)/\rho_r \). On any discrete memoryless channel, the quantity
\[ \left[ \sum_{\gamma_t \in S_t} e^{\alpha \Gamma(\gamma')} \right]^{1+\rho} \]
is bounded when \( \rho < \rho_r \) and \( 1/(1 + \rho_r) < \alpha < 1/(1 + \rho) \).

**Proof (0 < \rho < 1 only).** We use the general principles of Appendix A of Part II, but are forced to a still more refined version of Lemma A3, involving averaging over parts of a configuration.

We write
\[ \left[ \sum_{\gamma_t \in S_t} e^{\alpha \Gamma(\gamma')} \right]^{1+\rho} = \sum_{\gamma_1 \in S_t} e^{\alpha \Gamma(\gamma_1')} \left[ \sum_{\gamma_2 \in S_t} e^{\alpha \Gamma(\gamma_2')} \right]^\rho. \]
The pairs \((y_1', y_2')\) of incorrect paths, being members of the same incorrect subset of an infinite-constraint-length code, may be divided into configurations \(C_{\tau_1 \tau_2}^{\tau_0}y_1\), where \(\tau_1\) is the length of \(y_1'\), \(\tau_2\) the length of \(y_2'\), and \(\tau_0\) the number of time units over which they are merged. We say \(y_1' \in C_{\tau_1}\) if \(y_1''\) has length \(\tau_1\), and \(y_0' \in C_{\tau_0 \tau_1 \tau_2}^{\tau_0}y_1\) if \((y_1', y_0') \in C_{\tau_0 \tau_1 \tau_2}^{\tau_0}y_1\). Then we divide (83) into configurations and use Jensen's inequality, as usual, to get

\[
\sum_{y_1' \in S_1} e^{aR(y_1')} \left[ \sum_{y_2' \in S_1} e^{aR(y_2')} \right]^0
\]

\[
= \sum_{\tau_2 = 0}^{\infty} \sum_{y_1' \in C_{\tau_1}} e^{aR(y_1')} \left[ \sum_{\tau_2 = 0}^{\infty} \sum_{\tau_2 = 0}^{\min(\tau_1, \tau_2)} e^{aR(y_2')} \sum_{y_2' \in C_{\tau_0 \tau_1 \tau_2}^{\tau_0}y_1} e^{aR(y_2')} \right]^0
\]

\[
\leq \sum_{\tau_1 = 0}^{\infty} \sum_{\tau_2 = 0}^{\infty} \sum_{\tau_2 = 0}^{\min(\tau_1, \tau_2)} e^{aR(y_1')} \left[ \sum_{y_2' \in C_{\tau_0 \tau_1 \tau_2}^{\tau_0}y_1} e^{aR(y_2')} \right]^0, \quad 0 \leq \rho \leq 1.
\]

Let \(y_1'\) and \(y_2'\) be split into merged parts, \(y_1' = y_{10}, y_1'\), and unmerged parts \(y_{11}\) and \(y_{22}\). We first average only over the ensemble of choices of the unmerged path segments \(y_{22}\) (caret):

\[
e^{aR(y_1')} \left\{ \sum_{y_2' \in C_{\tau_0 \tau_1 \tau_2}^{\tau_0}y_1} e^{aR(G_{10} + G_{22})} \right\}^0
\]

\[
e^{aR(y_1')} \left[ \sum_{y_2' \in C_{\tau_0 \tau_1 \tau_2}^{\tau_0}y_1} e^{aR(y_2')} e^{aR(y_{22})} \right]^0
\]

\[
\leq e^{aR(y_1')} \left[ \sum_{y_2' \in C_{\tau_0 \tau_1 \tau_2}^{\tau_0}y_1} e^{aR(y_2')} e^{aR(y_{22})} \right]^0, \quad \rho \leq 1
\]

\[
= e^{aR(y_1')} \left[ \sum_{y_2' \in C_{\tau_0 \tau_1 \tau_2}^{\tau_0}y_1} e^{aR(y_{22})} \right]^0
\]

\[
= e^{aR(y_1')} e^{aR(y_{22})} \left[ \sum_{y_{22} = C_{\tau_0 \tau_1 \tau_2}^{\tau_0}y_1} e^{aR(y_{22})} \right]^0
\]

where we have recognized the independence of choices of \(y_{22}\) from \(y_1'\) and \(y_{10}\), and used \(\bar{R}^2 \leq R^2, \rho \leq 1\). Now we average over choices of \(y_1'\), which also fixes \(y_{20}\), and over the received word \(z\):

\[
e^{aR(y_1')} \left[ \sum_{y_2' = C_{\tau_0 \tau_1 \tau_2}^{\tau_0}y_1} e^{aR(y_{22})} \right]^0
\]

\[
\leq e^{a(1+\rho)R(G_{10})} e^{aR(G_{11})} \left[ \sum_{y_{22} = C_{\tau_0 \tau_1 \tau_2}^{\tau_0}y_1} e^{aR(y_{22})} \right]^0.
\]

(86)
By arguments similar to those in Lemma A3 of Part II we reduce this expression to

\[ | C_{\tau_1\tau_2}(y_1')|^\rho \exp -n\tau_0 E_a(\alpha, \rho) - n(\tau_1 - \tau_0) E_a(\alpha, \rho) - n(\tau_2 - \tau_0) E_c(\alpha, \rho), \]

where

\[ \exp -E_a(\alpha, \rho) = \sum_j \sum_{k'} q_j p_{k'} \left( \frac{p_{k'}}{q_j} \right)^{\alpha(1+\rho)}; \]

\[ \exp -E_b(\alpha, \rho) = \sum_j \sum_{k'} q_j p_{k'} \left( \frac{p_{k'}}{q_j} \right)^{\alpha}; \]

\[ \exp -E_c(\alpha, \rho) = \sum_j q_j \left[ \sum_{k'} p_{k'} \left( \frac{p_{k'}}{q_j} \right)^{\alpha} \right]^\alpha. \] (87)

We now use

\[ | C_{\tau_1} | \leq M_{\tau} \exp n\tau_0 r \exp n(\tau_1 - \tau_0)r; \]

\[ | C_{\tau_0\tau_1}(y_1') | \leq M_{\tau_2 - \tau_0} = \exp n(\tau_2 - \tau_0)r \]

to obtain

\[ \sum_{y_1' \in C_{\tau_1}} \sum_{y_2' \in C_{\tau_0\tau_1}(y_1')} e^{\alpha r(g_{y_1})} \left[ e^{\alpha r(g_{y_2})} \right]^\rho \leq \exp -n\tau_0[E_a(\alpha, \rho) - r] - n(\tau_1 - \tau_0)[E_b(\alpha, \rho) - r] - n(\tau_2 - \tau_0)[E_c(\alpha, \rho) - \rho r]. \] (89)

Finally, Hölder’s inequality yields

\[ \exp -[E_a(\alpha, \rho) - r] \leq \exp[1 - \alpha(1 + \rho)]r - \alpha(1 + \rho)E_0 \left[ \frac{1 - \alpha(1 + \rho)}{\alpha(1 + \rho)} \right] = \delta_a, \]

\[ 0 \leq \alpha(1 + \rho) \leq 1; \]

\[ \exp -[E_b(\alpha, \rho) - r] \leq \exp(1 - \alpha)r - \alpha E_0 \left( \frac{1 - \alpha}{\alpha} \right) = \delta_b, \]

\[ 0 \leq \alpha \leq 1; \]

\[ \exp -[E_c(\alpha, \rho) - \rho r] \leq \exp \rho(1 - \alpha)r - \rho \alpha E_0 \left( \frac{1 - \alpha}{\alpha} \right) = \delta_c, \]

\[ 0 \leq \rho \leq 1. \] (90)

For any \( \alpha < 1/1 + \rho \), or \( \rho < (1 - \alpha)/\alpha \), \( 0 \leq \rho \leq 1 \), all conditioning inequalities are satisfied and \( \delta_\alpha < 1 \) if

\[ \rho_\tau > \frac{1 - \alpha(1 + \rho)}{\alpha(1 + \rho)}, \] (91)
while $\delta_b$ and $\delta_c$ are less than 1 if

$$\rho_r > \frac{1 - \alpha}{\alpha} > \rho. \quad (93)$$

Clearly for $\rho \geq 0$, (93) implies (92). Consequently, for $0 \leq \rho \leq 1$, $\rho_r > (1 - \alpha)/\alpha > \rho$, we have

$$\left[ \sum_{Y \in S_r} e^{a \Gamma(Y)} \right]^{1-\rho} \leq \sum_{\tau_1=0}^{\infty} \sum_{\tau_2=0}^{\infty} \min(\tau_1, \tau_2) \delta_b^{\tau_1-\tau_2} \delta_e^{\tau_2-\tau_1} \delta_a^{\tau_1} \delta_b^{\tau_2} = K, \quad (94)$$

with $K$ some constant, since the sum can easily be shown to be convergent when $\delta_a$, $\delta_b$, and $\delta_c$ are all strictly less than 1.

APPENDIX B: "SEMISEQUENTIAL" LIST-OF-$L$ ALGORITHM

Viterbi (1967) suggested a "semisequential" maximum-likelihood decoding algorithm with some of the properties of sequential decoding. In this appendix we show that the parallel may be improved by using list-of-$L$ maximum-likelihood decoding.

Let $\rho_r$ satisfy $r = E_0(\rho_r)/\rho_r$, and let $L$ be an integer such that $\rho_r \leq L$. We indicated in Part II that the error probability for list-of-$L$ maximum-likelihood decoding of a constraint-length-$v$ code would be bounded by

$$\Pr_L(\Theta) \leq K \exp -nE_0(\rho_r) \quad (95)$$

(with a proof for $L \leq 2$).

Suppose now that the actual constraint length of the code is very long, effectively infinite, but that the decoder decodes as though the code had constraint length only $v$, retaining at any time $L$ survivors for each of the $M^v$ possible information sequences over the last $v$ time units. It is not hard to see that as long as the correct path survives, the probability of decoding error is still given by (95). However, should the correct path ever be rejected, with high probability all subsequent paths will look like garbage. For instance, with a biased metric $m_{jk} = \ln p_{jk}/q_j - r$, all paths will eventually (with probability 1) dip by $\beta$ for any $\beta$ if the true constraint length is infinite. Thus the probability of detecting a decoding error may be made 1, while, by increasing $\beta$, the probability of false alarm may be made as small as desired.

The semisequential algorithm then proceeds by starting to decode with a decoder suitable for a small $v$, and thus with complexity $G \sim LM^r$. If an error is detected, decoding is reinitiated with a larger $v$ at a point sufficiently
far back in time to be nearly certain of being before the error occurrence, where the correct path will have still been a survivor. The process is repeated with ever-increasing \( \nu \) as many times as necessary to get past the troublesome region. When the crisis seems to be over, the effective decoding constraint length can be prudently reduced again.

For definiteness assume that the constraint lengths \( \nu \) used are the integer multiples of some integer \( k \geq 1 \). The probability per unit time of having to employ a decoder of constraint length \( \nu + k \) is then the probability of list-of-\( L \) decoding error with decoding constraint length \( \nu \), given by (95). The probability of having to call on a decoder of complexity \( G \sim LM^{\nu k} \) is then bounded by (95), which with the substitution

\[
\nu \simeq \frac{\ln G/L}{\ln M} - k = \frac{\ln G/L}{nr} - k
\]

becomes

\[
Pr_L(\delta) \simeq K'G^{-\rho_r}
\]

using \( E_0(\rho_r)/\rho_r = r \), where

\[
K' \triangleq K L^\rho_r \exp knE_0(\rho_r).
\]

Thus we get the same Pareto distribution as with sequential decoding.

Two aspects of this result are worthy of note. First, we see a fundamental connection between list-of-\( L \) maximum-likelihood decoding and sequential decoding in the range of rates for which \( 1 < \rho_r \leq L \). This connection was also evidenced in the character of our proofs (for \( 1 < \rho_r \leq 2 \)). Second, as in our discussion of the Jacobs-Berlekamp result, we observe that selection of \( LM^\nu \) survivors at depth \( N_{\text{crit}} \) (where \( N_{\text{crit}} \) is the critical length at which decoding errors actually occur) may be regarded as a particular method of list-of-\( LM^\nu \)-decoding of the rate-\( r \) block code consisting of truncations of the infinite-\( \nu \) tree code to length \( N_{\text{crit}} \), and further that the block list decoding error probability of (97) is asymptotically optimum, as the reader may verify.

**Received:** November 21, 1972; **Revised:** January 15, 1974

**References**


CONVOLUTIONAL CODES. III


