that are hypothesized to be good ones. To carry out the evaluation, one finds weights of codewords corresponding to various input sequences $I(D)$. It would be advantageous to limit the length of those $I(D)$ sequences that could conceivably "achieve" free distance (by minimizing the right-hand side of (14)).

Theorem 3 shows that for noncatastrophic codes of rate $\frac{1}{2}$ there is no all-zero path of length $v - 1$ branches other than the path $\mathbf{0},\mathbf{0},\cdots,\mathbf{0}$. Hence any input $I(D)$ that does not induce the state $\mathbf{0}$ must have at least one nonzero output once every $v - 1$ blocks. Also the very first output block contains at least one nonzero output. Combining these results with the result of Theorem 4, we get Theorem 5.

Theorem 5: For noncatastrophic codes of rate $\frac{1}{2}$, free distance can be attained only by input sequences of length less than or equal to

$$(v + \lceil \log_2 v \rceil - 1)(v - 1) + 1.$$

Theorem 5 is an improvement on Costello's previous result [6], but the length is still of the order of v^2 and not $v \log v$, which was conjectured by Miczo and Rudolph [11].

References

[1] G. D. Forney, Jr., "Final report on a coding system design for advanced solar missions," NASA Arms Res. Ctr, Codex Corp., Watertown, Mass., Dec. 1967, Contract NAS2-3637, p. A20.

[2] J. K. Omura, "On the Viterbi decoding algorithm," *IEEE Trans. Inform. Theory* (Corresp.), vol. IT-15, Jan. 1969, pp. 177–179.

[3] J. L. Massey and M. K. Sain, "Inverses of linear sequential machines," *IEEE Trans. Comput.*, vol. C-17, Apr. 1968, pp. 330–337.

[4] J. J. Bussgang, "Some properties of binary convolutional code generators," *IEEE Trans. Inform. Theory*, vol. IT-11, Jan. 1965, pp. 90–100.

[5] S. Lin and H. Lyne, "Some results on binary convolutional code generators," *IEEE Trans. Inform. Theory* (Corresp.), vol. IT-13, Jan. 1967, pp. 134–139.

[6] D. J. Costello, Jr., "A construction technique for random-error-correcting convolutional codes," *IEEE Trans. Inform. Theory* (Corresp.), vol. IT-15, Sept. 1969, pp. 631–636.

[7] J. L. Massey, "Some algebraic and distance properties of convolutional codes," in *Error-Correcting Codes*, H. B. Mann, Ed. New York: Wiley, 1968, p. 90.

[8] J. A. Heller, "Sequential decoding: Short constraint length convolutional codes," Jet Propul. Lab., California Inst. Technol., Pasadena, Space Programs Summary 37-54, vol. 3, Dec. 1968, pp. 171–174.

[9] R. McEliece and H. C. Rumsey, "Capabilities of convolutional codes," Jet Propul. Lab., California Inst. Technol., Pasadena. Space Programs Summary 37–50, vol. 3, Apr. 1968, pp. 248–251,

[10] G. D. Forney, Jr., "Use of a sequential decoder to analyze convolutional code structure," *IEEE Trans. Inform. Theory* (Corresp.), vol. IT-16, Nov. 1970, pp. 793–795.

[11] A. Miczo and L. D. Rudolph, "A note on the free distance of a convolutional code," *IEEE Trans. Inform. Theory* (Corresp.) vol. IT-16, Sept. 1970, pp. 646–648.

[12] F. P. Preparata, "An improved upper bound to free distance for rate $1/n$ convolutional codes," in *Proc. UMR-Mervin J. Kelly Communication Conf.*, Rolla, Mo., 1970.

[13] J. Layland and R. McEliece, "An upper bound on free distance of a tree code," Jet Propul. Lab., California Inst. Technol., Pasadena, Space Program Summary, 37-62, vol. 3, Apr. 1970, pp. 63–64.

[14] J. P. Odenwalder, "Optimal decoding of convolutional codes," Ph.D. dissertation, Dep. Elec. Eng., Univ. California, Los Angeles, 1970, p. 47.

[15] W. J. Rosenberg, "Structural properties of convolutional codes," Ph.D. dissertation, Dep. Elec. Eng., Univ. California, Los Angeles, 1971, ch. 4.

[16] G. D. Forney, Jr., private communication.

Variable-Length Codes and the Fano Metric

JAMES L. MASSEY, FELLOW, IEEE

***Abstract*—It is shown that the metric proposed originally by Fano for sequential decoding is precisely the required statistic for minimum-error-probability decoding of variable-length codes. The analysis shows further that the "natural" choice of bias in the metric is the code rate and gives insight into why the Fano metric has proved to be the best practical choice in sequential decoding. The recently devised Jelinek–Zigangirov "stack algorithm" is shown to be a natural consequence of this interpretation of the Fano metric. Finally, it is shown that the elimination of the bias in the "truncated" portion of the code tree gives a slight reduction in average computation at the sacrifice of increased error probability.**

Manuscript received March 24, 1971. This work was supported by NASA Grant NGL 15-004-026 at the University of Notre Dame in liaison with the NASA Goddard Space Flight Center.

The author is with the Department of Electrical Engineering, University of Notre Dame, Notre Dame, Ind. 46556. He is now on leave of absence at the Laboratory for Communication Theory, Royal Technical University of Denmark, Lyngby, Denmark.

I. The Variable-Length Decoding Problem

CONSIDER the transmission situation shown in Fig. 1 for a variable-length code $\{x_1,x_2,\cdots,x_M\}$ whose codeword lengths are $\{n_1,n_2,\cdots,n_M\}$. The message m $(1 \leq m \leq M)$, having probability P_m, selects the codeword

$$x_m = [x_{m1},x_{m2},\cdots,x_{mn_m}]$$

to which is added the "random tail"

$$t_m = [t_1,t_2,\cdots,t_{N-n_m}]$$

to form the input sequence

$$z = [z_1,z_2,\cdots,z_N] = [x_m,t_m]$$

for transmission over the discrete memoryless channel (DMC). Here $N = \max(n_1,n_2,\cdots,n_M)$ is the maximum

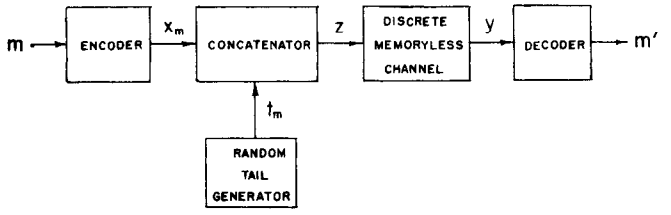

Fig. 1. Conceptual situation for the variable-length coding problem.

codeword length. We assume that t_m is selected statistically independently of x_m, and that the digits in t_m are chosen independently according to a probability measure $Q(\ )$ over the channel input alphabet; that is,

$$\Pr(t_m \mid x_m) = \Pr(t_m) = \prod_{k=1}^{N-n_m} Q(t_k).$$

The random tail t_m can be thought of as the digits resulting from subsequent encodings of further messages in a randomly selected code or simply as a convenient device for normalizing the number of received digits that must be considered in the decoding process.

Letting $y = [y_1, y_2, \cdots, y_N]$ be the received word, we have by the definition [1] of a DMC

$$\Pr(y \mid z) = \prod_{i=1}^{n_m} P(y_i \mid x_{mi}) \prod_{j=1}^{N-n_m} P(y_{n_m+j} \mid t_j),$$

where $P(\ |\ )$ defines the transition structure of the channel.

The joint probability of sending message m, adding the random tail t_m, and receiving y may thus be written

$$\begin{aligned}\Pr(m, t_m, y) &= P_m \Pr(t_m \mid x_m) \Pr(y \mid x_m t_m) \\ &= P_m \prod_{i=1}^{n_m} P(y_i \mid x_{mi}) \prod_{k=1}^{N-n_m} Q(t_k) \prod_{j=1}^{N-n_m} P(y_{n_m+j} \mid t_j).\end{aligned}$$

Summing over all possible random tails, we obtain

$$\Pr(m, y) = P_m \prod_{i=1}^{n_m} P(y_i \mid x_{mi}) \prod_{j=1}^{N-n_m} P_0(y_{n_m+j}), \qquad (1)$$

where

$$P_0(y_i) = \sum_{t_k} P(y_i \mid t_k) Q(t_k) \qquad (2)$$

is the probability measure induced on the channel output alphabet when the channel inputs are used according to $Q(\ )$. But given y, the *optimum* (in the sense of minimizing the probability of an erroneous decision) decoding rule is to choose m' as the value of m, which maximizes $\Pr(m, y)$ or equivalently which maximizes

$$\Pr(m, y) / \prod_{i=1}^{N} P_0(y_i)$$

since the denominator is independent of m. Taking logarithms, and using (1) and (2), we obtain as the final statistic to be maximized by the optimum decoder:

$$L(m, y) = \sum_{i=1}^{n_m} \left[\log \frac{P(y_i \mid x_{mi})}{P_0(y_i)} + \frac{1}{n_m} \log P_m \right]. \qquad (3)$$

We note the somewhat surprising fact that the statistic for each codeword depends only on that portion of the received word y having the same length as the codeword.

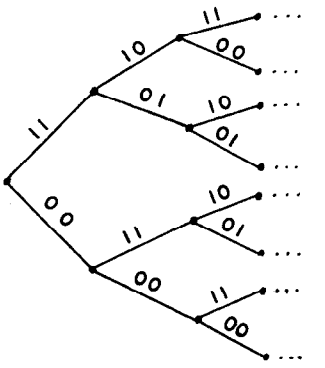

Fig. 2. An example of a tree code with rate $\frac{1}{2}$.

II. Application to Sequential Decoding

To simplify the discussion without loss of essential generality, we shall assume binary coding, i.e., we shall assume that the DMC is a binary input channel. Sequential decoding refers in general to a method for obtaining a good estimate of the path followed by the encoder of a tree code. An example of a (semi-infinite) tree code is shown in Fig. 2, where the encoder is supposed to follow the upper branch at each successive node if and only if the corresponding information digit is a "one." The code rate R is $\frac{1}{2}$ in this example, and more generally is the reciprocal of the number of encoded digits per information digit.

Now suppose that $\{x_1, x_2, \cdots, x_M\}$ represent all the paths in the encoding tree that have been explored up to the present by a sequential decoder. The decoder is assumed to know nothing about the digits in the unexplored part of the encoding tree except that they are selected independently according to $Q(\ )$, but for the price of one computation it can "buy" the knowledge of the digits on the branches stemming from the terminal nodes on any already explored path. Every sequential decoding algorithm can be thought of as a rule for deciding which of these paths to extend.

Assuming that the information bits are independent and equally likely to be zeros or ones, we have as the *a priori* probability that the encoder followed path x_m

$$P_m = 2^{-Rn_m}. \qquad (4)$$

We next recognize that the problem of deciding which of the explored paths is the initial portion of the path actually followed by the encoder is precisely the variable-length decoding problem of the previous section, since the set of already explored paths form a set of variable-length codewords one and only one of which was actually chosen by the

encoder. Hence the decoder should base its decision on the statistic $L(m,\mathbf{y})$, which with the use of (3) and (4) becomes

$$L(m,\mathbf{y}) = \sum_{i=1}^{n_m} \left[\log \frac{P(y_i \mid x_{mi})}{P_0(y_i)} - R\right]. \tag{5}$$

We now recognize $L(m,\mathbf{y})$ to be precisely the metric so brilliantly postulated on intuitive grounds by Fano [2] for sequential decoding. The quantity R on the right-hand side of (5) is called the *bias* in the metric. It is interesting to note that Gallager [1] suggests using R_{comp} as the bias whereas Jelinek [3] follows Fano in using R. Our analysis here shows that R is indeed the natural choice for the bias. It is also interesting to note that $L(m,\mathbf{y})$ has generally been thought of as a "law-of-large-numbers approximation" to the true path-likelihood functions [1], [3], [4], whereas our analysis shows that there is no approximation whatsoever.

In light of the above analysis, an obviously good sequential decoding rule would be to extend the explored path $\mathbf{x}_m$, which maximizes the Fano metric $L(m,\mathbf{y})$. In fact, this rule is precisely the so-called stack algorithm proposed independently by Jelinek [4] and Zigangirov [5]. By always choosing to extend the most likely explored path, one would expect that the average computation would be nearly minimized. Moreover, the original Fano algorithm [2] is essentially this same rule, since Geist [6] has shown that the first *new* node extended by this algorithm is the terminal node on the already explored path of greatest metric (within the quantization parameter Δ built into this algorithm).

III. Remark on Metrics for Finite Trees

In the usual practical case where the encoding tree has finite length, i.e., where L encoded digits result from branches corresponding to true information bits but T further digits are obtained by encoding zeros to terminate the code, (4) becomes

$$P_m = 2^{-R[\min(L,n_m)]}. \tag{6}$$

Thus, for $n_m > L$, the path-likelihood function (3) becomes

$$L(m,\mathbf{y}) = \sum_{i=1}^{L} \left[\log \frac{P(y_i \mid x_{mi})}{P_0(y_i)} - R\right] + \sum_{i=L+1}^{n_m} \left[\log \frac{P(y_i \mid x_{mi})}{P_0(y_i)}\right] \tag{7}$$

rather than (5), which suggests that the bias term R should be dropped for the digits in the truncated part of the encoding tree.

To test the validity of this suggestion, a Jelinek–Zigangirov decoder was used with a rate $\frac{1}{2}$ code on a binary symmetric channel (BSC) with $R_{\text{comp}} = R = \frac{1}{2}$, both with and without the bias term in the truncated part of the tree. The results are given in Table I, and show that there is indeed the expected improvement in computation when the bias is removed. However, the improvement is very slight and is paid for by a factor of two increase in decoder undetected error probability. Richer [7] has reported similar results for decoding of a rate $\frac{1}{12}$ code. We conclude that in practice it would be generally unwise to remove the bias in the truncated portion of the tree unless the slight improvement in computation was badly needed, or unless $T \gg L$ so that most of the computation is done in the truncated part of the tree and thus removing the bias there would significantly speed up the decoding process.

TABLE I

Effect of Removing Bias in Truncated Part of Encoding Tree for a Frame of 256 Information Bits Encoded With the $R = \frac{1}{2}$ Convolutional Code With Generators (Octal) 400,000,000,000 and 651,102,104,421 [35 Branches in Truncated Part of Tree] on the BSC With Crossover Probability 0.045 [$R = R_{\text{comp}}$]

N	Number of Frames Out of 1000 Decoded Frames With Computation N or Less	
	With Usual Bias in Truncated Part of Tree	Without Usual Bias in Truncated Part of Tree
320	147	148
340	360	364
360	486	489
400	629	635
500	794	801
600	856	860
900	933	938
1200	958	961
Erased frames	42	39
Erroneously decoded frames	28	58

Finally, we wish to remark that although the Jelinek–Zigangirov algorithm always extends the most likely explored path and hence maximizes the probability that the next step taken is along the correct path in the encoding tree, it does not follow that this algorithm strictly minimizes the average computation to find the correct path over the ensemble of randomly chosen unexplored parts of the encoding tree. The guaranteed optimality is for the next step only. Determination of the conditions for which this algorithm actually minimizes the average computation and determination of the general decoding rule for minimizing average computation remain as interesting open problems.

References

[1] R. G. Gallager, *Information Theory and Reliable Communication*. New York: Wiley, 1968.
[2] R. M. Fano, "A heuristic discussion of probabilistic decoding," *IEEE Trans. Inform. Theory*, vol. IT-9, Apr. 1963, pp. 64–73.
[3] F. Jelinek, *Probabilistic Information Theory*. New York: McGraw-Hill, 1968.
[4] ——, "A fast sequential decoding algorithm using a stack," *IBM J. Res. Dev.*, vol. 13, Nov. 1969, pp. 675–685.
[5] K. Sh. Zigangirov, "Some sequential decoding procedures," *Probl. Peredach. Inform.*, vol. 2, 1966, pp. 13–25.
[6] J. M. Geist, "Algorithmic aspects of sequential decoding," Dep. Elec. Eng., Univ. Notre Dame, Notre Dame, Ind., Tech. Rep. EE-702, Aug. 1970.
[7] I. Richer, private communication, M.I.T. Lincoln Lab., Lexington, Mass., Jan. 1971.