Universal Decoding for Finite-State Channels

JACOB ZIV, FELLOW, IEEE

Abstract—**Universal decoding procedures for finite-state channels are discussed. Although the channel statistics are not known, universal decoding can achieve an error probability with an error exponent that, for large enough block length (or constraint length in case of convolutional codes), is equal to the random-coding error exponent associated with the optimal maximum-likelihood decoding procedure for the given channel. The same approach is applied to sequential decoding, yielding a universal sequential decoding procedure with a cutoff rate and an error exponent that are equal to those achieved by the classical sequential decoding procedure.**

I. Introduction

CONSIDER THE CLASS of finite alphabet time-discrete channels for which the output depends on the current input letter and the current channel state, and where the current state is a function of the previous state and the previous input and output letters of the channel. (When there is only a single possible state the channel is memoryless.)

It is well known that if random coding is used, it is possible to achieve an error probability that vanishes exponentially with the block length (or "constraint length"), provided that the channel statistics are known and that an optimal (maximum-likelihood) decoder is used.

In many cases of interest, the channel statistics are not known and, therefore, the optimal maximum-likelihood decoder cannot be utilized.

A universal decoding procedure is described. Although the channel statistics are not known, this universal decoding procedure achieves an error probability with an error exponent that is equal to the one associated with random coding followed by an optimal, maximum-likelihood decoding procedure for block codes as well as for convolutional codes. The universal decoding procedure is based on a parsing procedure where the "tested" message and the received message are parsed jointly. A universal distance measure, which is a function of the number of parsed "phrases," is sequentially generated for the tested message. The decoded message is the one with the smallest "distance" from the received message.

A universal version of the sequential decoding procedure for random convolutional codes is derived, yielding an error probability with an error exponent equal to the one associated with the classical sequential decoding where the channel statistics are fully known. Furthermore, the computational complexity is comparable to that of the classical sequential decoder.

In Section II the main results are stated. The discussion in this section is concerned with block codes. In Section III the results of Section II are extended to convolutional codes. In section IV a universal version of the sequential decoding procedure for convolutional codes is derived. In the Appendix some properties of the proposed universal distance "measure" are stated and then used (Theorem 2) in the proof of the main theorem of Section II.

II. Statement of Main Results

Consider the class of finite-alphabet, finite-state channels characterized by a transition probability distribution of the form

$$W(\boldsymbol{u}|\boldsymbol{x}) = \prod_{i=1}^{n} W(y_i|x_i, s_i), \tag{1}$$

where

- x_i input to the channel at the ith instant; $x_i \in X$, $|X| = \alpha$;
- $\boldsymbol{x}$ $= x_1, x_2, \cdots, x_i, \cdots, x_n = x_1^n$;
- y_i output of the channel at the ith instant; $y_i \in Y$, $|Y| = \beta$;
- s_i state of the channel at the ith instant; $s_i \in S$, $|S| = K$;
- s_i $= q(y_{i-1}, x_{i-1}, s_{i-1})$ where q is the next-state function and where s_1 is the initial state (some member of S). The channel is memoryless if $K = 1$.

Consider a code $C = \{\boldsymbol{x}^1, \boldsymbol{x}^2, \cdots, \boldsymbol{x}^l, \cdots, \boldsymbol{x}^M\}$ for $M = 2^{nR}$ equiprobable messages, where $\boldsymbol{x}^l \in X^n$; $l = 1, 2, \cdots, M$. It is well known that the optimal decoding procedure that minimizes the probability of error is the maximum-likelihood decoding procedure where the decoded codeword is an $\boldsymbol{x}' \in C$ for which

$$f(\boldsymbol{x}', \boldsymbol{y}) = \min_{\boldsymbol{x} \in C} f(\boldsymbol{x}, \boldsymbol{y}), \tag{2}$$

where the function $f(\boldsymbol{x}, \boldsymbol{y})$ is given by

$$f(\boldsymbol{x}, \boldsymbol{y}) = -\log(W(\boldsymbol{y}|\boldsymbol{x})). \tag{3}$$

(Throughout this paper $\log(\cdot)$ means $\log_2(\cdot)$.)

The probability of error associated with this optimal decoding procedure is denoted by $P_{e,0}(C, R, n)$.

Manuscript received October 13, 1983; revised November 19, 1984. This work was done in part while on sabbatical leave at Bell Telephone Laboratories, Murray Hill, NJ 07974, USA. The work was supported by the US–Israel Binational Science Foundation. The material in this paper was partially presented at Globcom '82, Miami, FL, November 29 through December 2, 1982.

The author is with the Department of Electrical Engineering, Technion—Israel Institute of Technology, Haifa, Israel, and currently on leave with AT & T Bell Laboratories, Murray Hill, NJ 07974, USA.

0018-9448/85/0700-0453$01.00 ©1985 IEEE

A. Random Coding

In general, for large $M = 2^{nR}$ it is very hard to find the optimal code that minimizes $P_{e,0}(C, R, n)$ over all codes with 2^{nR} codewords. However, it has been demonstrated [1] that the expectation of $P_{e,0}(C, R, n)$ over the ensemble of randomly selected codes where each codeword is generated independently at random, governed by some probability distribution $q(\boldsymbol{x})$, has the following interesting properties. Letting the expectation of $P_{e,0}(C, R, n)$ with respect to $q(\cdot)$ be denoted by $\overline{P}_e(q, R, n)$,

1) For memoryless channels ($K = 1$) there exists an $R(q)$ such that

$$-\frac{1}{n}\log \overline{P}_e(q, R, n) \geq E(q, R) > 0, \qquad R < R(q). \tag{4}$$

Hence, the expectation of the probability of error decreases exponentially with n for $R < R(q)$. Furthermore,

$$\max_{q(\cdot)}\left[-\frac{1}{n}\log \overline{P}_e(q, R, n)\right] = -\lim_{n\to\infty}\frac{1}{n}\log\left\{\min_{C} P_{e,0}(C, R, n)\right\},$$
$$\text{for } R_C \leq R \leq R_{\max}, \tag{5}$$

where $R_{\max}$ is the channel capacity. The rate R_C is called the critical rate [1].

2) For finite-state channels (i.e., $K > 1$)

$$-\frac{1}{n}\log \overline{P}_e(q, R, n) \geq E(q, R) > 0,$$
$$\text{for } R < R(q), \tag{6}$$

where now $\max_q R(q)$ is not necessarily equal to $R_{\max}$.

B. Universal Decoding

In some applications the channel's transition probability distribution $W(\boldsymbol{y}|\boldsymbol{x})$ is not available. In such cases maximum-likelihood decoding cannot be utilized and the optimizing channel-input distribution $q(\cdot)$ is not known. It is therefore assumed that $q(\cdot)$ is a uniform probability distribution, i.e., that $q(\boldsymbol{x}) = 1/|B|$ for every input vector $\boldsymbol{x} \in B$, where $B \subset X^n$ is the set of allowable n- dimensional input vectors. The simplest, most common case is $B \equiv X^n$, and therefore $q(\boldsymbol{x}) = \alpha^{-n}$.

In this paper, a universal decoding procedure is described that, for any finite-state channel, yields essentially the same random-coding error exponent as that associated with the optimal maximum-likelihood procedure. In this section, we deal with the universal decoding of block codes. In Section III the universal decoding of convolutional codes for finite-state channels is discussed. A universal sequential-decoding procedure for finite-state channels is discussed in Section IV.

In this section and in Section III the universal decoding rule that replaces the maximum-likelihood decoding rule is as follows (compare with (2) and (3)). The decoded message is $\boldsymbol{x}' \in C$, for which

$$f(\boldsymbol{x}', \boldsymbol{y}) = \min_{\boldsymbol{x}\in C} f(\boldsymbol{x}, \boldsymbol{y}), \tag{7}$$

where now, in contrast with (3),

$$f(\boldsymbol{x}, \boldsymbol{y}) = u(\boldsymbol{x}, \boldsymbol{y}), \tag{8}$$

where $u(\cdot, \cdot)$ is a universal function, i.e., a function of $\boldsymbol{x}$ and $\boldsymbol{y}$ that is *independent* of $W(\boldsymbol{y}|\boldsymbol{x})$.

Let us denote by $M_0(\boldsymbol{x}, \boldsymbol{y})$ the cardinality of the set

$$S_0(\boldsymbol{x}, \boldsymbol{y}) = \left\{\boldsymbol{x}'\colon \boldsymbol{x}, \boldsymbol{x}' \in B;\ W(\boldsymbol{y}|\boldsymbol{x}') \geq W(\boldsymbol{y}|\boldsymbol{x})\right\}. \tag{9}$$

Also let $M_u(\boldsymbol{x}, \boldsymbol{y})$ denote the cardinality of the set

$$S_u(\boldsymbol{x}, \boldsymbol{y}) = \left\{\boldsymbol{x}'\colon \boldsymbol{x}, \boldsymbol{x}' \in B;\ u(\boldsymbol{x}', \boldsymbol{y}) \leq u(\boldsymbol{x}, \boldsymbol{y})\right\}. \tag{10}$$

Denote by $\overline{P}_{e,0}(R, n)$ the random-coding average error probability for the case where $q(\boldsymbol{x}) = 1/|B|$ for all $\boldsymbol{x} \in B$ and where the optimal maximum-likelihood decoding rule is used ((2) and (3)). Similarly, denote by $\overline{P}_{e,u}(R, n)$ the random-coding average error probability for the case where $q(\boldsymbol{x}) = 1/|B|$ for all $\boldsymbol{x} \in B$ and where the *universal* decoding rule (7) and (8)) is applied. Then, we have the following.

Theorem 1:

$$\overline{P}_{e,0}(R, n) = 1 - E\left(1 - M_0(\boldsymbol{x}, \boldsymbol{y})\frac{1}{|B|}\right)^{2^{nR}-1} \tag{11}$$

$$\overline{P}_{e,u}(R, n) = 1 - E\left(1 - M_u(\boldsymbol{x}, \boldsymbol{y})\frac{1}{|B|}\right)^{2^{nR}-1}, \tag{12}$$

where $E(\cdot)$ denotes expectation over $(\boldsymbol{X} \times \boldsymbol{Y})$ with respect to $P(\boldsymbol{x}, \boldsymbol{y}) = q(\boldsymbol{x})W(\boldsymbol{y}|\boldsymbol{x})$.

Corollary 1:

$$\overline{P}_{e,0}(R, n) \leq \overline{P}_{e,u}(R, n)$$
$$\leq 2\overline{P}_{e,0}(R, n)\left[\max_{\boldsymbol{x}\in B,\ \boldsymbol{y}\in Y^n}\frac{M_u(\boldsymbol{x}, \boldsymbol{y})}{M_0(\boldsymbol{x}, \boldsymbol{y})} + 1\right]. \tag{13}$$

The proof of the corollary appears in the Appendix.

Proof of Theorem 1: Let us assume that $\boldsymbol{x}$ is the transmitted codeword and that $\boldsymbol{y}$ is the received n vector. It follows from (2) and (3) that when the maximum-likelihood decoding procedure is applied, $\boldsymbol{x}$ will be correctly decoded if and only if none of the other $2^{nR} - 1$ codewords (which are all picked at random and independently with a uniform distribution over B) is an element in $S_0(\boldsymbol{x}, \boldsymbol{y})$ (see (9)). (A tie $W(\boldsymbol{y}|\boldsymbol{x}) = W(\boldsymbol{y}|\boldsymbol{x}')$ is counted as an error). Thus, the average probability of being correct, given $\boldsymbol{x}$ and $\boldsymbol{y}$, is equal to

$$P_{c,0}(\boldsymbol{x}, \boldsymbol{y}) = \left[1 - \frac{M_0(\boldsymbol{x}, \boldsymbol{y})}{|B|}\right]^{2^{nR}-1}, \qquad (14)$$

where $M_0(\boldsymbol{x}, \boldsymbol{y}) = |S_0(\boldsymbol{x}, \boldsymbol{y})|$. Hence

$$\begin{aligned}\overline{P}_{e,0}(R,n) &= E(1 - P_{c,0}(\boldsymbol{x}, \boldsymbol{y}))\\ &= 1 - E\left[1 - \frac{M_0(\boldsymbol{x}, \boldsymbol{y})}{|B|}\right]^{2^{nR}-1}, \qquad (15)\end{aligned}$$

where $E(\cdot)$ denotes expectation with respect to $P(\boldsymbol{x}, \boldsymbol{y}) = (1/|B|)W(\boldsymbol{y}|\boldsymbol{x})$. In a similar way,

$$\overline{P}_{e,u}(R,n) = 1 - E\left[1 - \frac{M_u(\boldsymbol{x}, \boldsymbol{y})}{|B|}\right]^{2^{nR}-1}. \qquad (16)$$

Discussion: It will be demonstrated that for the important case where $B = X^n$ as well as for the case where B is the set of all n vectors with some prescribed composition, there exists a universal decoding function $u(\boldsymbol{x}, \boldsymbol{y})$ such that for any finite-state channel

$$\max_{\boldsymbol{x} \in B:\ \boldsymbol{y} \in Y^n} \left[\frac{M_u(\boldsymbol{x}, \boldsymbol{y})}{M_0(\boldsymbol{x}, \boldsymbol{y})}\right] \le 2^{\epsilon(n,K)n}, \qquad (17)$$

where $\lim_{n\to\infty}\epsilon(n,K) = 0$ for any finite K. (K is the number of channel states). Hence, by (17) and Corollary 1 (13)),

$$\frac{1}{n}\log \overline{P}_{e,u}(R,n) = \frac{1}{n}\log \overline{P}_{e,0}(R,n) + \delta(n,K), \qquad (18)$$

where $\lim_{n\to\infty}\delta(n,K) = 0$ for any finite K. Thus both the optimal $\overline{P}_{e,0}(R,n)$ and the "universal" $\overline{P}_{e,u}(R,n)$ have essentially the same error exponent.

In the following the main result of this section is derived. A new universal decoding function is introduced, and it is demonstrated that this decoding function is asymptotically optimal in the sense that

$$\boxed{\lim_{n\to\infty}\frac{1}{n}\log \overline{P}_{e,0}(R,n) = \lim_{n\to\infty}\frac{1}{n}\log \overline{P}_{e,u}(R,n),}$$

for any finite-state channel.

C. *The Universal Decoding Function $u(\boldsymbol{x}, \boldsymbol{y})$ for Finite-State Channels*

Given $\boldsymbol{x}$ and $\boldsymbol{y}$, consider the sequence $\boldsymbol{w}$ of ordered pairs

$$\boldsymbol{w} = w_1 w_2 \cdots, \qquad w_i = x_i, y_i.$$

For the incremental parsing [3] of $\boldsymbol{w}$ into g strings (phrases), 1) all phrases (except for the last one) must be distinct, and 2) the prefix of each phrase (i.e., the last symbol of the phrase is deleted) is identical with some previous phrase. For example, let

$$\boldsymbol{x} = 0 \mid 1 \mid 0 \quad 0 \mid 0 \quad 1 \mid$$

$$\boldsymbol{y} = 0 \mid 1 \mid 0 \quad 1 \mid 0 \quad 1 \mid$$

$$\boldsymbol{w} = (0,0)(1,1)(00,01)(01,01), \qquad g = 4.$$

Thus

$$\boldsymbol{w} = w_1 w_2^{l_2} w_{l_2+1}^{l_3} w_{l_3+1}^{l_4} \cdots w_{l_{g-1}+1}^{n}, \qquad (19)$$

where

$$\begin{aligned} w_i^j &= w_i w_{i+1} \cdots w_j\\ w_{l_i+1}^{l_{(i+1)}} &\ne w_{l_k+1}^{l_{(k+1)}}, \qquad \text{for all } k,\ 0 \le k < i < g\\ w_{l_i}^{l_{(i+1)}-1} &= w_{l_k+1}^{l_{(k+1)}}, \qquad \text{for some } k,\ 0 \le k < i \le g\\ l_0 &\triangleq 0\\ l_1 &\triangleq 1.\end{aligned}$$

Clearly, all the phrases are distinct (except, perhaps, for the last one). The number of distinct phrases, $c(\boldsymbol{x}, \boldsymbol{y})$, is therefore at least $g - 1$. Now, let $c(\boldsymbol{y})$ be the number of distinct phrases in the parsed $\boldsymbol{y}$ and let $\boldsymbol{y}(l)$ denote the lth distinct $\boldsymbol{y}$ phrase $(1 \le l \le c(\boldsymbol{y}))$.

In the above example there are three distinct $\boldsymbol{y}$ phrases (namely $0, 1, 01$), while $c(\boldsymbol{x}, \boldsymbol{y}) = 4$.

Let $c_l(\boldsymbol{x}|\boldsymbol{y})$ be the number of times that the $\boldsymbol{y}$ phrase $\boldsymbol{y}(l)$ appears in the parsed $\boldsymbol{y}$. Clearly, $c_l(\boldsymbol{x}|\boldsymbol{y})$ counts the number of distinct $\boldsymbol{x}$ phrases that appear jointly with $\boldsymbol{y}_l$ and $\sum_{l=1}^{c(\boldsymbol{y})} c_l(\boldsymbol{x}|\boldsymbol{y}) = c(\boldsymbol{x}, \boldsymbol{y})$. In the above example

$$\begin{array}{lll} \boldsymbol{y}(1) = 0 & \boldsymbol{y}(2) = 1 & \boldsymbol{y}(3) = 01\\ c_1(\boldsymbol{x}|\boldsymbol{y}) = 1 & c_2(\boldsymbol{x}|\boldsymbol{y}) = 1 & c_3(\boldsymbol{x}|\boldsymbol{y}) = 2. \end{array}$$

Now, let the universal decoding function be

$$u(\boldsymbol{x}, \boldsymbol{y}) = \frac{1}{n}\sum_{l=1}^{c(\boldsymbol{y})} c_l(\boldsymbol{x}|\boldsymbol{y}) \log c_l(\boldsymbol{x}|\boldsymbol{y}). \qquad (20)$$

Theorem 2: For any finite-state channel and for $B = X^n$ or for the case where $B = T_P$ for some given P_x

$$\lim_{n\to\infty}\frac{1}{n}\log M_u(\boldsymbol{x}, \boldsymbol{y}) = \lim_{n\to\infty}\frac{1}{n}\log M_0(\boldsymbol{x}, \boldsymbol{y}), \qquad (21)$$

and therefore by (14) (Corollary 1)

$$\lim_{n\to\infty}\frac{1}{n}\log \overline{P}_{e,u}(R,n) = \lim_{n\to\infty}\frac{1}{n}\log \overline{P}_{e,0}(R,n). \qquad (22)$$

The proof of Theorem 2 is given in the Appendix.

It should be noted again that the new universal decoding function $u(\boldsymbol{x}, \boldsymbol{y})$ is asymptotically optimal for *any* finite K, unlike other known universal decoding functions that were derived for the *special case* of memoryless channels (i.e., $K = 1$) as shown below.

D. *A Universal Decoding Function $u_1(\boldsymbol{x}, \boldsymbol{y})$ for Memoryless Channels*

We proceed now to study a universal decoding function for the special case where $K = 1$ (memoryless channels), following Csiszár and Körner [2]. As in [2], the "type" of a sequence $\boldsymbol{x} \in X^n$ is the distribution P_x on X where $P_x(a)$ is the relative frequency of $a \in X$ in $\boldsymbol{x}$. The set of sequences of type P_x is denoted by T_P. Similarly, for every $\boldsymbol{x} \in X^n$ any $\boldsymbol{y} \in Y^n$, if $(\boldsymbol{x}, \boldsymbol{y})$ has a joint type

$$P_{x,y}(a,b) = P_x(a)V(b|a) = Q_y(b)U(a|b),$$

where Q_y is the type of y, x has conditional type U given y, and y has conditional type V given x. The set of all x of conditional type U given y is denoted by $T_{U(y)}$. The set of all y of conditional type V given x is denoted by $T_V(x)$.

Lemma 1: Assume that the transmitted message x is of conditional type U (i.e., $x \in T_U(y)$) and that $y \in T_Q$. Then, for memoryless channels (i.e., $K = 1$) and for $B = X^n$ or for the case where $B = T_P$ for some given P_x

$$M_0(x, y) \geq 2^{nH(V|Q)}(n+1)^{-\alpha\beta}. \tag{23a}$$

Furthermore, for $f(x', y) = u_1(x', y) = H(U'|Q)$

$$\lim_{n\to\infty} \frac{1}{n} \log M_{u_1}(x, y) = \lim_{n\to\infty} \frac{1}{n} \log M_0(x, y), \tag{23b}$$

where $H(U|Q)$ is the conditional entropy

$$H(U|Q) = -\sum_{b\in Y}\sum_{\alpha\in X} Q_y(b)U(a|b)\log U(\alpha|b). \tag{24}$$

Proof: x is in $T_U(y)$. Therefore, any other $x' \in T_U(y)$ has the property that $x' \in T_P$ if $x \in T_P$, and that

$$W(y|x) = \prod_{i=1}^{n} W(y_i|x_i) = \prod_{i=1}^{n} W(y_i|x_i') = W(y|x'),$$

where $W(y|x)$ is the n-dimensional transition probability of the channel.

Thus [2]

$$M_0(x, y) = |\{x' \colon W(y|x') \geq W(y|x)|$$
$$\geq |T_U(y)| \geq (n+1)^{-\alpha\beta_2 nH(U|Q)}.$$

Following [2], let us substitute in (8) $f(x', y) \triangleq u_1(x', y) = H(U'|Q)$ and let $B = X^n$ or $B = T_P$ for some P_x. Then by [2, eqs. (2.5.1) and (2.5.2)] and by (10)

$$M_{u_1}(x, y) = |\{x' \colon H(U|Q) \geq H(U'|Q)\}|$$
$$\leq 2^{nH(U|Q)}(n+1)^{\alpha\beta}, \tag{25}$$

where $(n+1)^{\alpha\beta}$ is an upper bound on the number of possible empirical distributions U' over $X \times Y$.

Comparing (25) with (23a) yields

$$\lim_{n\to\infty} \frac{1}{n} \log M_{u_1}(x, y) = \lim_{n\to\infty} \frac{1}{n} \log M_0(x, y).$$

Hence, by (13) (Corollary 1) the universal decoding function $u_1(x', y)$ is asymptotically optimal for all discrete memoryless channels ($K = 1$) in the sense that $\lim_{n\to\infty}(1/n)P_{e,u_1}(R, n) = \lim_{n\to\infty}(1/n)P_{e,0}(R, n)$, as demonstrated by Csiszár and Körner [2] for the case where $B = T_P$ (using another bounding technique). However, $u_1(x', y) = H(U'|Q)$ is not optimal for the more general case where the channel is not memoryless (i.e., $K > 1$).

It should be noted that Csiszár and Körner also demonstrated in [2] the existence of a *single universal code* that can be used independently of the actual channel statistics and still yield an error exponent identical to the random-coding exponent for any discrete memoryless channel when universally decoded as above.

III. Universal Decoding of Convolutional Codes

In this section it will be demonstrated that the universal decoding function $u(x', y)$, which was introduced in the previous section, can replace the optimal, maximum-likelihood decoding function $f(x', y) = -\log W(y|x')$ for decoding convolutional codes, yielding a random-coding error exponent that is essentially equal to the error exponent associated with the optimal maximum-likelihood decoding procedure (for large constraint length).

The reader is referred to [4] for definitions of convolutional (trellis) codes and results concerning the random-coding error exponent that is associated with maximum-likelihood (Viterbi) decoding of convolutional codes for memoryless channels. It is shown in [4, eq. (5.1.12)] that the probability of error at any node of a convolutional code tree is bounded by

$$P_e(j) < \sum_{k=0}^{\infty} \pi_k(j), \tag{26}$$

where

$$\pi_k(j) = P_r\left\{\begin{matrix}\text{error caused by any one of up}\\ \text{to } 2^k \text{ incorrect paths un-}\\ \text{merged from node } j \text{ to node}\\ j+k+N\end{matrix}\right\}$$

and where N is the "constraint length" [4]. Furthermore, by [4, eq. (5.1.16)] and by (26)

$$\bar{P}_e(j) \leq \sum_{k=u}^{\infty} \bar{\pi}_k(j) \leq \sum_{k=0}^{\infty} \bar{P}_e\left(\frac{k}{N+k}R, (N+k)l\right), \tag{27}$$

where: 1) $(\bar{\cdot})$ denotes expectation over the ensemble of randomly selected convolutional codes (with an independent and identically distributed (iid), uniform distribution on X); 2) l is the number of channel symbols per branch; 3) $\bar{P}_e(R, n)$ is the average probability of error for random block codes of length n and rate R. Equations (26) and (27) hold for any decoding function. It therefore follows from the results of Section II that, for any finite-state channel, if the optimal maximum-likelihood function $f(x_j(k), y_j(k))$ is replaced by the universal $u(x_j(k), y_j(k))$ (or by $u_1(x_j(k), y_j(k))$, if $K = 1$), where $x_j(k) = x_{j+1}^{k} = x_{j+1}, x_{j+2}, \cdots, x_k$, (following the notation of [4],) then the upper bound on the ensemble average of the probability of error will have the same exponent (for large N) as that which is associated with the optimal maximum-likelihood decoding ([4], eq. (5.1.21)).

IV. Universal Decoding Version of Sequential Decoding

The universal decoding function $u(x, y)$ was shown to be optimal for large block length n or constraint lengths N.

In practice, one well-known scheme that can deal with large constraint length, while keeping the decoder complexity reasonably small, is sequential decoding [4]. Sequential decoding seems to be superior to Viterbi decoding (i.e., maximum-likelihood decoding) of convolutional codes (in terms of the average per/letter probability of error) for a given decoding complexity if the decoding delay is not of prime importance [4, p. 378]. On the other hand, sequential decoding seems to be less robust than the Viterbi decoding [4, p. 378].

It will be demonstrated that by replacing the classical sequential-decoding metric, which is channel dependent, with a universal decoding metric (which is channel independent), it is possible to make the sequential-decoding procedure universal and therefore *totally insensitive* to variation in the channel parameters, without any deterioration in the achieved error exponent (for large N), or in the computational cutoff rate.

Let $m(\boldsymbol{x}_j(k))$ denote the metric attached to an input vector $\boldsymbol{x}_j(k) = x_{j+1}, \cdots, x_k$ upon observing $\boldsymbol{y}$. In the classical case [4], for memoryless channels,

$$m(\boldsymbol{x}_j(k)) = \sum_{i=j+1}^{k} m(x_i) = \sum_{i=j+1}^{k} \left(\log \frac{P(y_i|x_i)}{P(y_i)} - R \right). \tag{28}$$

If the channel statistics are not known, the metric of (28) cannot be used.

Let $m_u(\boldsymbol{x}_j(k))$ denote a universal metric for $\boldsymbol{x}_j(k)$, which is independent of the channel statistics, defined as follows:

$$m_u(\boldsymbol{x}_j(k)) = l(k-j)\{-u(\boldsymbol{x}_j(k), \boldsymbol{y}_j(k)) + \log \alpha - (R + \Delta)\}, \tag{29}$$

where $u(\cdot,\cdot)$ is given by (23) (or, if the channel is known to be memoryless, also by $u(\cdot,\cdot) = u_1(\cdot,\cdot)$ as in (23)), l is the number of channel symbols per branch, and Δ is an arbitrarily small positive number. It should be noted that, unlike the classical $m(\boldsymbol{x}_j(k))$ of (28), the universal metric $m_u(\boldsymbol{x}_j(k))$ is not additive, i.e., in general,

$$m_u(\boldsymbol{x}_j(k)) \neq m_u(\boldsymbol{x}_j(i)) + m_u(\boldsymbol{x}_i(k)). \tag{30}$$

The sequential-decoding stack algorithm which is described in [4] has to be modified accordingly. The ordering of the tested paths in the stack is done in the following way. Assume that $\boldsymbol{x}_0(k)$ and $\tilde{\boldsymbol{x}}_0(t)$ are two tested paths in the stack and assume that their first j branches are identical, i.e., $\hat{\boldsymbol{x}}_0(j) = \tilde{\boldsymbol{x}}_0(j)$.

Then $\hat{\boldsymbol{x}}_0(k)$ will be above $\tilde{\boldsymbol{x}}_0(t)$ in the stack if

$$\max_{\boldsymbol{x}'_j(k):\ \boldsymbol{x}'_0(j+1) = \hat{\boldsymbol{x}}_0(j+1)} m_u(\boldsymbol{x}'_j(k)) > \max_{\boldsymbol{x}'_j(t):\ \boldsymbol{x}'_0(j+1) = \tilde{\boldsymbol{x}}_0(j+1)} m_u(\boldsymbol{x}'_j(t)), \tag{31}$$

where $\boldsymbol{x}'_0(k)$ are elements of the stack, and $\boldsymbol{x}'_j(k)$ consists of the last $(k-j)$ branches of $\boldsymbol{x}'_0(k)$. For example, in Fig. 1, let $j = 0$ and let $m_u(\boldsymbol{x}'_0(1)) = 4$, $m_u(\boldsymbol{x}''_1(2)) = 2$, $m_u(\boldsymbol{x}''_0(2)) = 0$, $m_u(\boldsymbol{x}'''_0(2)) = 3$ and $m_u(\boldsymbol{x}'''_1(2)) = 1$. Then $\boldsymbol{x}'$ is placed above both $\boldsymbol{x}''$ and $\boldsymbol{x}'''$ since

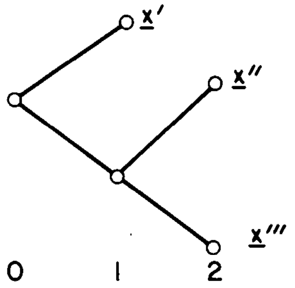

Fig. 1. Decoding tree—example.

$$m_u(\boldsymbol{x}'_0(1)) > \max\{m_u(\boldsymbol{x}''_0(2)), m_u(\boldsymbol{x}'''_0(2))\} = 3.$$

Now, both $\boldsymbol{x}''$ and $\boldsymbol{x}'''$ will be below $\boldsymbol{x}'_0(1)$, but x'' will be above $\boldsymbol{x}'''$ although $m_u(x'''_0(2)) > m_u(x'''_0(3))$ since

$$m_u(\boldsymbol{x}''_1(2)) > m_u(\boldsymbol{x}'''_1(2)).$$

Thus, the top path is always the one with the largest metric, but the one below it is not necessarily the path with the second largest metric. Observe that the ordering according to (31) coincides with the standard stack ordering if the additive metric of (28) is to be used.

This modified ordering of (31) is slightly more complicated than the classical one since whenever a path of length L branches in the stack is extended to the length of $(L+1)$ branches, there are $(L+1)$ different metrics that should be evaluated (rather than only one in the standard case). As in [4] let us denote by $\boldsymbol{x}$ the correct path through the trellis, and let $\boldsymbol{x}'_j$ be some incorrect path that diverges from $\boldsymbol{x}$ at node j (i.e., $\boldsymbol{x}'_j$ is an element of the jth incorrect subset of paths in the stack [4]).

In the following the expected number of computations (defined to be the expected number of tested paths per node) as well as the average probability of error will be evaluated for the ensemble of randomly selected convolutional codes with an iid, uniformly generating probability distribution.

A. The Expected Number of Computations

As in [4, eq. (6.2.1)] the incorrect path $\boldsymbol{x}'_j$ may be searched beyond the node $k > j$ only if

$$m_u(\boldsymbol{x}'_j(k)) \geq \min_{i \geq j} m_u(\boldsymbol{x}_j(i)) \triangleq \gamma_j. \tag{32}$$

The number of computations in the jth node is defined to be the number of elements in the jth subset denoted by C_j. The expected number of computations $\overline{C}_j$ is therefore bounded by

$$\overline{C}_j \leq \sum_{k=j}^{\infty} 2^{l(k-j)R} P_r\left\{m_u\left((\boldsymbol{x}'_j(k)) \geq \gamma_j\right)\right\}.$$

Now,

$$P_r\left\{m_u\left(x_j'(k)\right) \geq \gamma_j)\right\} \leq \sum_{i>j} P_r\left\{m_u\left(x_j'(k)\right) \geq m_u\left(x_j(i)\right)\right\}.$$

Thus

$$\bar{C}_j \leq \sum_{k=j}^{\infty} 2^{l(k-j)R} \sum_{i>j} P_r\left\{m_u\left(x_j'(k)\right) \geq m_u\left(x_j(i)\right)\right\}. \tag{33}$$

Also, by (29)

$$P_r\left\{m_u\left(x_j'(k)\right) \geq m_u\left(x_j(i)\right)\right\} = P_r\left\{u\left(x_j'(k), y_j(k)\right) \leq D_i\right\}, \tag{34}$$

where

$$D_i = \left\{l(i-j)\left(u\left(x_j(i), y_j(i)\right) - \log\alpha + (R+\Delta)\right) + l(k-j)(\log\alpha - (R+\Delta))\right\} \frac{1}{l(k-j)}.$$

Now, let $N_u(D_i)$ denote the number of vectors $x_j'(k) \in X^{l(k-j)}$ for which $u(x_j'(k), y_j(k)) \leq D_i$. Then, by Lemma 2 (see the Appendix)

$$N_u(D_i) \leq 2^{l(k-j)[D_i + \epsilon(k-j)]}, \tag{35}$$

where

$$\epsilon(k-j) = 0\left(\frac{\log\log(k-j)}{\log(k-j)}\right).$$

Therefore,

$$P_r\left[u\left(x_j'(k), y_j(k)\right) < D_i | x, y\right] = N_u(D_i) 2^{-(k-j)l\log\alpha} \leq 2^{l(k-j)[D_i + \epsilon(k-j) - \log\alpha]}, \tag{36}$$

where $2^{-l(k-j)\log\alpha}$ is the probability measure of each of the $N_u(D_i)$ vectors. Thus, by (34) and (36)

$$P_r\left[u\left(x_j'(k), y_j(k)\right) \leq D_i | x, y\right] \leq 2^{l(k-j)\epsilon(k-j)} 2^{(i-j)l(u(x_j(i), y_j(i)) - \log\alpha + (R+\Delta)) - (k-j)l(R+\Delta)}. \tag{37}$$

Therefore, by (34) and (37)

$$P_r\left\{m_u\left(x_j'(k)\right) \geq m_u\left(x_j(i)\right)\right\} \leq 2^{l(k-j)\epsilon(k-j)} E\left\{2^{(i-j)l\{u(x_j(i), y_j(i)) - \log\alpha\}} 2^{-(k-j)l(R+\Delta)}\right\}, \tag{38}$$

where $E\{\cdot\}$ denotes expectation over $\{x_j(i), y_j(i)\}$. Now, by (11) for the case where $M = 2^{nR} = 2$

$$\bar{P}_{e,0}\left(\frac{1}{n}, n\right) = EM_0(x, y) 2^{-n\log\alpha}.$$

Also, by (A.5)

$$M_0(x, y) \geq 2^{n\{u(x, y) - 0(1/\log n)\}}.$$

(If the channel is memoryless the same expression also holds for $u_1(x, y)$.) Thus

$$\bar{P}_{e,0}\left(\frac{1}{n}, n\right) \geq E 2^{n\{u(x, y) - \log\alpha\}} 2^{-n0(1/\log n)}. \tag{39}$$

Applying (39) to (38) yields

$$P_r\left\{m_u\left(x_j'(k)\right) \geq m_u\left(x_j(i)\right)\right\} \leq 2^{l(k-j)\epsilon(k-j)} 2^{(i-k)l(R+\Delta)} \bar{P}_{e,0}\left(\frac{1}{(i-j)l}, (i-j)l\right) \cdot 2^{+l(i-j)0(1/\log(i-j)l)}. \tag{40}$$

Now, let

$$R_0 \triangleq \min_n \frac{1}{n}\left(-\log\bar{P}_{e,0}\left(\frac{1}{n}, n\right)\right). \tag{41}$$

If the channel is memoryless then R_0 is the zero-rate error exponent of the random-coding error bound [4] (for the case where the randomizing iid distribution is uniform over X). Therefore, by (40) and (41)

$$P_r\left\{m_u\left(x_j'(k)\right) \geq m_u\left(x_j(i)\right)\right\} \leq 2^{l(k-j)\epsilon(k-j)} 2^{l(i-j)\lfloor (R+\Delta) - R_0 + \delta(i-j) \rfloor} 2^{-l(k-j)(R+\Delta)}, \tag{42}$$

where $\delta(i-j) = 0(1/\log(i-j)l)$. Inserting (42) into (33) yields

$$\bar{C}_j \leq \sum_{n=1}^{\infty} 2^{(R - R_0 + \Delta + \delta(n))nl} \sum_{m=1}^{\infty} 2^{-(\Delta - \epsilon(m))lm} \leq D(R, \Delta) \tag{43}$$

for $R < R_0 - \Delta$, where $D(R, \Delta)$ is a finite constant that is independent of the coding constraint length N for any rate $R < R_0 - \Delta$ and where Δ is an arbitrarily small positive number.

Hence, R_0 is the computational complexity cutoff rate, below which the average number of computations is bounded regardless of how large the constraint length is. For the memoryless case, the cutoff rate is *identical* to that of the classical sequential-decoding stack algorithm. Furthermore, (42) holds also for the more general finite-state case, and that the modified stack algorithm discussed here is *universal* and is therefore robust to variations in the channel statistics. Clearly, by the Chebyshev inequality,

$$P_r\left[C_j > L\right] \leq \frac{D(R, \Delta)}{L}, \qquad R < R_0 - \Delta \tag{44}$$

(compare with [4, eq. (6.2.22)]).

B. An Upper Bound on the Error Probability

An error may be caused by selecting an incorrect path $x_j'(k)$ that diverges from the correct path at node j and remerges with it at node k only if

$$m_u\left(x_j'(k)\right) > \min_{i>j} m_u\left(x_j(i)\right) \triangleq \gamma_j.$$

Therefore, it follows from [4, p. 363] that the average error probability at node j is

$$\overline{P}_e(j) \leq \sum_{k=j+N}^{\infty} 2^{l(k-j-N)R} P_r\left[m_u\left(\boldsymbol{x}_j'(k)\right) \geq \gamma_j\right]$$

$$< \sum_{k=j+N}^{\infty} 2^{l(k-j-N)R}$$

$$\cdot \sum_{i>j} P_r\left[m_u\left(\boldsymbol{x}_j'(k)\right) > m_u\left(\boldsymbol{x}_j(i)\right)\right]. \tag{45}$$

Hence, by (42), (43), and (44),

$$\overline{P}_e(j) \leq \overline{C}_j 2^{NlR} \leq D(R, \Delta) 2^{Nb}, \qquad R < R_0 - \Delta,$$

where $b = lR$ is the number of input bits per branch (compare with [4], (6.3.12)). Thus, for memoryless channels the error exponent of the modified universal stack algorithm is essentially equal to the exponent associated with the classical algorithm. Furthermore, (45) holds for channels with finite memory as well.

Acknowledgment

The author wishes to acknowledge with thanks helpful discussions with J. Körner, J. K. Wolf, and A. D. Wyner.

Appendix

Proof of Corollary 1

Let $P_{e,0}(\boldsymbol{x}, \boldsymbol{y}) = 1 - P_{c,0}(\boldsymbol{x}, \boldsymbol{y})$ and $P_{e,u}(\boldsymbol{x}, \boldsymbol{y}) = 1 - P_{c,u}(\boldsymbol{x}, \boldsymbol{y})$. Then, by Eq. (16)

$$P_{e,u}(\boldsymbol{x}, \boldsymbol{y})$$

$$= P_{e,0}(\boldsymbol{x}, \boldsymbol{y}) \frac{P_{e,u}(\boldsymbol{x}, \boldsymbol{y})}{P_{e,0}(\boldsymbol{x}, \boldsymbol{y})}$$

$$\leq P_{e,0}(\boldsymbol{x}, \boldsymbol{y})$$

$$\cdot \frac{|B|^{-1}(2^{nR}-1) M_u(\boldsymbol{x}, \boldsymbol{y})}{|B|^{-1}(2^{nR}-1) M_0(\boldsymbol{x}, \boldsymbol{y}) - \frac{1}{2}|B|^{-2}(2^{nR}-1)^2 M_0^2(\boldsymbol{x}, \boldsymbol{y})}$$

since

$$(1-a)^n > 1 - na$$

and

$$(1-a)^n < 1 - na + \frac{n(n-1)}{2} a^2 < 1 - na + \frac{n^2}{2} a^2.$$

Thus

$$P_{e,u}(\boldsymbol{x}, \boldsymbol{y}) \leq P_{e,0}(\boldsymbol{x}, \boldsymbol{y}) \frac{M_u(\boldsymbol{x}, \boldsymbol{y})}{M_0(\boldsymbol{x}, \boldsymbol{y})}$$

$$\cdot \frac{1}{1 - \frac{1}{2}|B|^{-1}(2^{nR}-1) M_0(\boldsymbol{x}, \boldsymbol{y})}. \tag{A.1}$$

Now, let

$$S_1 = \left\{(\boldsymbol{x}, \boldsymbol{y}) \colon M_0(\boldsymbol{x}, \boldsymbol{y}) \leq |B|(2^{nR}-1)^{-1}\right\}.$$

Then, by (A.1) and (18)

$$\begin{aligned}\overline{P}_{e,u}(R, n) &= EP_{e,u}(\boldsymbol{x}, \boldsymbol{y}) \\ &\leq P_r(S_1) E_{S_1}(P_{e,u}(\boldsymbol{x}, \boldsymbol{y})) + 1 - P_r(S_1) \\ &\leq 2 \max_{\boldsymbol{x} \in B, \boldsymbol{y} \in Y^n} \left(\frac{M_u(\boldsymbol{x}, \boldsymbol{y})}{M_0(\boldsymbol{x}, \boldsymbol{y})}\right) P_r(S_1) E_{S_1}(P_{e,0}(\boldsymbol{x}, \boldsymbol{y})) \\ &\quad + 1 - P_r(S_1),\end{aligned} \tag{A.2}$$

where $E_{S_1}(\cdot)$ denotes expectation, conditioned on the event that $(\boldsymbol{x}, \boldsymbol{y}) \in S_1$.

Now,

$$P(S_1) E_{S_1}(P_{e,0}(\boldsymbol{x}, \boldsymbol{y})) \leq \overline{P}_{e,0}(R, n). \tag{A.3}$$

Also, by Eq. (11)

$$\overline{P}_{e,0}(R, n) \geq (1 - P_r(S_1))\left[1 - \left(1 - (2^{nR}-1)^{-1}\right)^{(2^{nR}-1)}\right].$$

Thus, since

$$\left(1 - \frac{1}{a}\right)^{\alpha} < \frac{1}{2}, \qquad a = 2^{nR} - 1,$$

then

$$\overline{P}_{e,0}(R, n) \geq (1 - P_r(S_1)) \frac{1}{2},$$

and therefore

$$1 - P_r(S_1) \leq 2\overline{P}_{e,0}(R, n). \tag{A.4}$$

Inserting (A.3) and (A.4) into (A.2) yields

$$\overline{P}_{e,u}(R, n) \leq \overline{P}_{e,0}(R, n)\left[2 \max_{B, Y^n}\left(\frac{M_u(\boldsymbol{x}, \boldsymbol{y})}{M_0(\boldsymbol{x}, \boldsymbol{y})}\right) + 2\right].$$

Proof of Theorem 2

Lemma 1:

$$\begin{aligned}\log M_0(\boldsymbol{x}, \boldsymbol{y}) &\geq \sum_{l=1}^{c(\boldsymbol{y})} \log c_l(\boldsymbol{x}|\boldsymbol{y}) - 0\left(\frac{n}{\log n}\right) \log K^2 \\ &= n\left[u(\boldsymbol{x}, \boldsymbol{y}) - 0\left(\frac{1}{\log n}\right) \log K^2\right].\end{aligned} \tag{A.5}$$

(See Section II for definitions of $c_l(\boldsymbol{x}|\boldsymbol{y})$, $c(\boldsymbol{y})$, $u(\boldsymbol{x}, \boldsymbol{y})$.)

The (optimal) maximum likelihood decoder computes $f(\boldsymbol{x}, \boldsymbol{y}) = -\log W(\boldsymbol{y}|\boldsymbol{x})$ for each one of the codewords. This can be done by a finite-state machine F defined by a six-tuple (S, X, Y, D, g, q), where S is a finite set of states, X and Y are the two input alphabets, and D is a set of output letters. The function g is the next-state function that maps $S \times X \times Y$ into S. The function q is the output function that maps $S \times X \times Y$ into S.

The machine is fed with the codeword $\boldsymbol{x}$ and the received vector $\boldsymbol{y}$ and emits the sequence $\boldsymbol{d} = d_1, d_2, \cdots, d_i, \cdots, d_n$ while going through a sequence of states $\boldsymbol{s} = s_1, s_2, \cdots, s_i, \cdots, s_n$; $s_i \in S$ according to

$$\begin{aligned}d_i &= -\log W(y_i|x_i, s_i) \\ s_{i+1} &= g(s_i, x_i, y_i).\end{aligned} \tag{A.6}$$

Clearly,

$$f(\boldsymbol{x}, \boldsymbol{y}) = \sum_{i=1}^{n} d_i. \tag{A.7}$$

Consider now the incremental parsing $\boldsymbol{w}$ of $(\boldsymbol{x}, \boldsymbol{y})$ (see (19)) and the set $\boldsymbol{P}(\boldsymbol{y}, s, s', l)$ of $\boldsymbol{x}$ vectors that are generated from $\boldsymbol{x}$ by permuting x phrases among the $c_l(x|y,s,s')$ distinct x phrases of length l that appear jointly with y_l and that have an initial state s and a final state s'. Then, by (A.6), (A.7), and (1), $f(x',y)=f(x,y)$ for every $x' \in P(y,s,s',l)$. Hence, by (9), (19), and (A.5),

$$M_0(x,y) \geq \prod_{l=1}^{c(y)} \prod_{(s,s')} c_l(x|y,s,s')! \tag{A.8}$$

Thus, by the Stirling formula

$$\begin{aligned}\log M_0(x,y) &> \sum_{l=1}^{c(y)} \sum_{s,s'} c_l(x|y,s,s')\left[\log c_l(x|y,s,s') - \log e\right] \\ &= -\sum_{l=1}^{c(y)} c_l(x|y) \sum_{s,s'} \frac{c_l(x|y,s,s')}{c_l(x|y)} \log \frac{c_l(x|y)}{c_l(x|y,s,s')} \\ &\quad + \sum_{l=1}^{c(y)} c_l(x|y)(\log c_l(x|y) - \log e)\end{aligned}$$

since

$$\sum_{s,s'} c_l(x|y,s,s') = c_l(x|y).$$

Therefore, by the convexity of the logarithmic function

$$\begin{aligned}\log M_0(x,y) &\geq \sum_{l=1}^{c(y)} c_l(x|y)(\log c_l(x|y) - \log K^2 - \log e) \\ &= \sum_{l=1}^{c(y)} c_l(x|y) \log c_l(x|y) - c(x,y) \log K^2 e.\end{aligned} \tag{A.9}$$

Now, by [3], $c(x,y) \leq \sum_{i=1}^{j}(\alpha\beta)^i$, where j is the smallest integer for which $\sum_{i=1}^{j} i(\alpha\beta)^i \geq n$.

Thus

$$c(x,y) \leq 0\left(\frac{n}{\log n}\right). \tag{A.10}$$

Therefore, by (A.9) and (A.10)

$$\begin{aligned}\log M_0(x,y) &\geq \sum_{l=1}^{c(y)} c_l(x|y) \log c_l(x|y) - 0\left(\frac{n}{\log n}\right) \log K^2 \\ &= n\left[u(x,y) - \frac{1}{n} 0\left(\frac{n}{\log n}\right) \log K^2\right]\end{aligned}$$

Lemma 2: The number of sequences $x' \in X^n$ such that $u(x',y) \leq D$ is no more than $2^{n[D+0((\log\log n)/\log n)]}$. (Obviously, by (25) Lemma 2 holds also for $u_1(x',y)$).

Proof of Lemma 2: Consider the incremental parsing of (x',y) according to (19). Let us encode each phrase by an encoding algorithm based on an incremental parsing of $w=(x',y)$ that yields $c(w)$ *distinct* phrases (the last phrase of w generated by incremental parsing might be not new). The vector y is considered to be known at the receiver, and it is assumed that all the earlier phrases have already been decoded. Therefore, the prefix $w_{l_i+1}^{l_{(i+1)}-1}$ of the current phrase $w_{l_i+1}^{l_{(i+1)}}$ (resulting from the deletion of the last symbol w_{l_i+1} of the current phrase) can be regenerated at the decoder by pointing to the *one* earlier phrase that is identical to this prefix. This is done by first informing the decoder of the length of this prefix, denoted by $L(y(l))$, where $y(l)$ is the y phrase associated with the prefix $w_{l_i+1}^{l_{(i+1)}-1}$, namely y_{l_i+1}. Once the $L(y(l))$ is known, $y(l)$ itself is known. (Since y is available at the receiver!) The decoder is then informed of the serial number of the one earlier phrase of length $L(y(l))$ that is identical with the prefix $w_{l_i+1}^{l_{(i+1)}-1}$ among the $c_l(x'|y)$ phrases which are characterized by the same associated y phrase $y(l)$.

Once the prefix of the current phrase is regenerated at the decoder, the last x letter $x'_{l_{(i+1)}}$ of the phrase $w_{l_i+1}^{l_{(i+1)}}$ is regenerated by informing the decoder of its value. The total length of the codeword (given y) is thus upper bounded by

$$\begin{aligned}L(x'|y) \leq \sum_{l=1}^{c(y)} (c_l(x'|y))\{\log(c_l(x'|y)+1) \\ + \log\alpha + 2\log L(y(l)) + 4\}.\end{aligned} \tag{A.11}$$

Now, by the convexity of the logarithmic function

$$\sum_{l=1}^{c(y)} c_l(x'|y) \log L(y(l)) \leq c(y) \log \frac{n}{c(y)}.$$

Also, as in (A.10),

$$c(y) \leq 0\left(\frac{n}{\log n}\right).$$

Thus

$$\frac{1}{n}\sum_{l=1}^{c(y)} (c_l(x'|y)+1) \log L(y(l)) \leq 0\left(\frac{\log\log n}{\log n}\right). \tag{A.12}$$

Therefore, for this information lossless coding scheme we have, by (20), (A.11), and (A.12), that

$$L(x'|y) \leq n\left\{u(x',y) + 0\left(\frac{\log\log n}{\log n}\right)\right\}.$$

If $u(x',y) \leq D$ it follows that

$$L(x'|y) \leq n\left[D + 0\left(\frac{\log\log n}{\log n}\right)\right].$$

Hence, the number of vectors x' such that $u(x',y) \leq D$ is upper bounded by

$$2^{n[D+0((\log\log n)/\log n)]}.$$

Now, by Lemma 2 and (10)

$$\begin{aligned}M_u(x,y) &\triangleq |\{x': x' \in X^n;\ u(x',y) \leq u(x,y)\}| \\ &\leq 2^{[u(x,y)+0(\log\log n/\log n)]}.\end{aligned} \tag{A.13}$$

Thus, by (A.5) and (A.12)

$$M_u(x,y) \leq M_0(x,y) 2^{\epsilon(K,n)n},$$

where $\lim_{n\to\infty}\epsilon(K,n)=0$ for any finite K. This completes the proof of Theorem 2.

References

[1] R. G. Gallager, *Information Theory and Reliable Communication.* New York: Wiley, 1968.

[2] I. Csiszár and J. Körner, *Information Theory: Coding Theorems for Discrete Memoryless Systems.* New York: Academic, 1981.

[3] J. Ziv and A. Lempel, "Compression of individual sequences via variable-rate coding" *IEEE Trans. Inform. Theory*, vol. IT-24, no. 5, pp. 530–536, Sep. 1978.

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