Coding Theorems for Individual Sequences

JACOB ZIV, FELLOW, IEEE

Abstract—**A quantity called the *finite-state* complexity is assigned to every infinite sequence of elements drawn from a finite set. This quantity characterizes the largest compression ratio that can be achieved in accurate transmission of the sequence by any finite-state encoder (and decoder). Coding theorems and converses are derived for an individual sequence without any probabilistic characterization, and universal data compression algorithms are introduced that are asymptotically optimal for all sequences over a given alphabet. The finite-state complexity of a sequence plays a role similar to that of entropy in classical information theory (which deals with probabilistic ensembles of sequences rather than an individual sequence). For a probabilistic source, the expectation of the finite state complexity of its sequences is equal to the source's entropy. The finite state complexity is of particular interest when the source statistics are unspecified.**

I. Introduction

OUR PROBLEM concerns the system shown in Fig. 1. The sequence u consists of letters drawn from an alphabet of α letters occuring at the rate ρ symbols/s. The sequence v is the encoded version of u and consists of letters drawn from an alphabet of β letters at rate ρ. The sequence $\hat{u}$ is the decoded version of v and should be an accurate replica of u.

Let the density of errors be defined by

$$d(u,\hat{u}) = \limsup_{n\to\infty} \frac{1}{n} D(u_1^n, \hat{u}_1^n)$$

where $D(u_1^n, \hat{u}_1^n)$ is the Hamming distance between the n-vectors $u_1, u_2, \cdots, u_n$ and $\hat{u}_1, \hat{u}_2, \cdots, \hat{u}_n$. In data compression the aim is to minimize β while keeping $d(u,\hat{u})$ negligible. That is, we require $d(u,\hat{u}) \leqslant \epsilon$ where ϵ is an arbitrary small positive number.

Restricting the discussion to finite-state encoders and decoders, we shall define a finite-state complexity $H(u)$ of an individual sequence u. Coding theorems and their converses are then derived, which demonstrate that $H(u)$ is equal to the minimum of $\log_2\beta$ over all finite-state encoders and decoders such that $d(u,\hat{u}) \leqslant \epsilon$ for an arbitrary small $\epsilon > 0$. Furthermore, the coding theorem demonstrates the existence of an asymptotically optimal universal block encoding scheme that achieves an arbitrary small distortion $d(u,\hat{u})$, as the block length approaches infinity, for *all* sequences such that $H(u) \leqslant \log_2\beta$. This finite-state complexity of an *individual* sequence (without any probabilistic characterization) therefore, plays a role similar to that of entropy in classical information theory, which deals with probabilistic ensembles of sequences rather than with individual sequences.

Manuscript received September 24, 1976; revised October 7, 1977.

The author is with Bell Laboratories, Murray Hill, NJ 07974, on leave from the Department of Electrical Engineering, Technion—Israel Institute of Technology, Haifa, Israel.

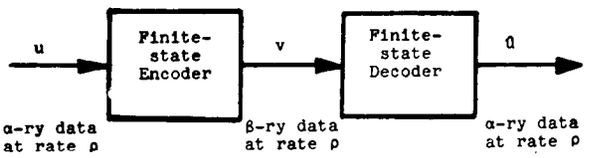

Fig. 1. Fixed-rate finite-state data compression system.

It is demonstrated that $H(u)$ is lower bounded by the Lempel–Ziv complexity [1]. Finally, it is shown that a universal data compression algorithm recently introduced in [2] is asymptotically optimal and has an implementation complexity that grows only linearly with the block length.

In Section II, we give a formal statement of the problem and state the main results, and in Section III we give the proofs. In Section IV, we conclude with some observations regarding possible generalizations.

II. Formal Statement of the Problem and Results

Let U be a set of α symbols that we call the source alphabet. Elements in U are called letters.

Input sequence: Consider the infinite sequence

$$u = u_1, u_2, \cdots, \qquad u_i \in U$$

and let

$$u_i^j = u_i, u_{i+1}, \cdots, u_j.$$

u is called the *input* or *source* sequence.

Encoder: An encoder is a mapping from the space of all infinite sequences u to the space of all infinite sequences v of letters drawn from an alphabet V of β letters. The sequence v is called the *encoded sequence*. A *block encoder* is a mapping

$$v_{i+1}^{i+n} = b\left(u_{i+1}^{i+n}\right), \qquad i = 0, n, 2n, \cdots$$

where $b(\cdot)$ is a function that maps n-vectors of letters drawn from U (α-letters) into n-vectors drawn from V (β-letters).

A *casual sliding block encoder* is the mapping [11]

$$v_i = l\left(u_{i-n+1}^{i}\right), \qquad i \geqslant n$$

where $l(\cdot)$ is a function that maps n-vectors of letters drawn from U into letters drawn from V.

Both these encoders (as well as noncausal sliding-block encoders) are members of the more general class of finite-state encoders defined as follows.

0018-9448/78/0700-405$00.75

An encoder is a *finite-state encoder* $c[s_1, S, f(\cdot,\cdot), g(\cdot,\cdot)]$ if for some positive integer t we have

$$v_{i-t} = f(u_i, s_i), \qquad i > t$$
$$s_i = g(u_{i-1}, s_{i-1}), \qquad i \geq 2 \tag{1}$$

where s_i, the state of the encoder at the ith instant, is one out of S states and is a function of u_{i-1} and the previous state s_{i-1}. s_1 is the initial state, and t is called the *coding delay*.

Examples:

1) A *causal sliding block encoder*

$$s_i = u_{i-n+1}^{i-1} = g(u_{i-1}, s_{i-1}) = g(u_{i-n}^{i-1}),$$
$$u_i = u_1, \text{ for } 0 \geq i \geq -n,$$
$$S = \alpha^{n-1} \qquad t = 0$$
$$v_i = f(s_i, u_i) = f(u_{i-n}^{i}).$$

2) A *block encoder*

$$s_i = \left(u_{i-2n}^{i-1}, \quad i - \left\lfloor \frac{i}{n} \right\rfloor n\right), \qquad u_i = u_1, \text{ for } 0 \geq i \geq -2n$$

(where $\lfloor x \rfloor$ denotes the largest integer that is smaller than or equal to x)

$$S = n\alpha^{2n} \qquad t = n$$
$$v_{i-t} = f(u_i, s_i) = f\left(u_{i-2n}^{i}; i - \left\lfloor \frac{i}{n} \right\rfloor n\right).$$

Observation: When a finite-state encoder is used, any l-vector u_i^{i+l-1} may be mapped into at most one out of S possible l-vectors of letters drawn from V, since there are S possible states s_i at the ith instant.

Decoder: A decoder is a mapping of the infinite sequence v into the sequence $\hat{u}$, where $\hat{u}_i \in U$. A *finite-state decoder* $d[s_1, S, q(\cdot,\cdot), k(\cdot,\cdot)]$ is defined as follows:

$$\hat{u}_{i-t} = q(v_i, s_i), \qquad i > t,$$
$$s_i = k(v_{i-1}, s_{i-1}), \qquad i \geq 2. \tag{2}$$

s_i is the state of the decoder at the ith instant and is one out of S possible states. s_1 is the initial state, and t is the decoding delay.

In all that follows the discussion is limited to the class of finite-state encoders and decoders.

The aim of data compression is to make β as small as possible while keeping $\hat{u}$ an accurate replica of u. Two cases will be considered:

Case I: $\hat{u} \equiv u$,

Case II: $d(u,\hat{u}) \triangleq \limsup_{n\to\infty} \frac{1}{n} D(u_1^n, \hat{u}_1^n) \leq \epsilon$,

where ϵ is an arbitrary small positive number and $D(u_1^n, \hat{u}_1^n)$ is the Hamming distance between u_1^n and $\hat{u}_1^n$, i.e.,

$$D(u_1^n, \hat{u}_1^n) = \sum_{i=1}^{n} D(u_i, \hat{u}_i), \qquad D(u_i, \hat{u}_i) = \begin{cases} 0, & u_i = \hat{u}_i \\ 1, & u_i \neq \hat{u}_i. \end{cases}$$

Thus some errors are tolerated, provided that the error-rate is arbitrarily small. (More general fidelity criteria will be discussed elsewhere—see also the remark in Section IV.)

Results:

1) *Case I:* $\hat{u} \equiv u$: Let $2^{lh_l(u)}$ be the number of distinct l-vectors that are contained in u. (That is, by sliding a "window" of length l along u, we count the number of distinct l-vectors that appear through that window.

The existence of the limit $h(u) \equiv \lim_{l\to\infty} h_l(u)$, which plays an important role throughout this paper, is demonstrated in Section III, wherein the following coding theorem and its converse are proved.

Theorem 1 (Converse to Coding Theorem):

$$\hat{u} \neq u, \qquad \text{if } h(u) > \log_2 \beta$$

Theorem 2 (Coding Theorem): For any $n \geq \alpha$, there exists a block encoder (and a corresponding block decoder)

$$v_{i+1}^{i+n} = f(u_{i+1}^{i+n}), \quad \hat{u}_{i+1}^{i+n} = g(v_{i+1}^{i+n}), \qquad i = 0, n, 2n, \cdots$$

such that $\hat{u} = u$ for *all* sequences u for which

$$h(u) \leq \log_2 \beta - \epsilon_n(u)$$

where

$$\lim_{n\to\infty} \epsilon_n(u) = 0.$$

More precisely, let l be the largest integer such that $l^2\alpha^l \leq n$. Then

$$\epsilon_n(u) \leq \left(1 + \frac{l}{n}\right) h_l(u) - h(u) + \frac{4}{l} \log_2 \alpha.$$

2) *Case II:* $d(u,\hat{u}) \leq \epsilon$ for any arbitrary positive ϵ.

This is the case in which some errors are tolerated in order to achieve even greater compression than in Case I. However, one must keep the error-rate as arbitrarily small.

For any two infinite sequences u and w such that

$$u = u_1, u_2, \cdots, \qquad u_i \in U,$$
$$w = w_1, w_2, \cdots, \qquad w_i \in U,$$

let

$$d(u,w) = \limsup_{n\to\infty} \frac{1}{n} D(u_1^n, w_1^n). \tag{3}$$

Also let

$$H_d(u) \triangleq \inf_{w:\, d(u,w) < d} h(w) \tag{4}$$

and

$$H(u) \triangleq \lim_{d\to 0} H_d(u) \triangleq \sup_{d>0} H_d(u). \tag{5}$$

Clearly $H(u) \leq h(u)$. For example, let u be a typical infinite sequence from an i.i.d. source with unequal and nonzero letter probabilities. For such a sequence $h(u) = \log_2 \alpha$, since the relative frequency of any l-sequence will be nonzero. But $H(u) = H < \log_2 \alpha$ (see Theorem 5). The following coding theorem and its converse are proved in Section III.

Theorem 3 (Converse to Coding Theorem): If

$$H(u) > \log_2 \beta$$

then there exists an $\epsilon>0$ such that $d(u,\hat{u})>\epsilon$ for all finite-state encoder-decoder pairs.

Theorem 4 (Coding Theorem): For any $n \geqslant \alpha$ and $\delta>0$, there exists a block encoder and a block decoder

$$v_{i+1}^{i+n}=f(u_{i+1}^{i+n}), \quad \hat{u}_{i+1}^{i+n}=g(u_{i+1}^{i+n}), \qquad i=0,n,2n,\cdots$$

such that $d(u,\hat{u}) \leqslant 2\delta$ for *all* sequences such that

$$H(u)<\log_2\beta-\epsilon_n(u,\delta)-\delta$$

where $\lim_{n\to\infty}\epsilon_n(u,\delta)=0$.

Discussion: It is clear from the coding theorems and their converses that $H(u)$, which is defined for each individual infinite sequence, plays a role similar to that of Shannon's entropy (which is defined only for an ensemble of sequences with a stationary probability measure) in the sense that $H(u)$ corresponds to the smallest $\log_2\beta$ for which the error rate (or the probability of error in the classical probabilistic case) can be made to approach zero [7], [13].

The following theorem is proved in Section III.

Theorem 5: If u is drawn from an ergodic source with entropy H, then $H(u)=H$ almost surely. That is, $P[|H(u)-H|=0]=1$.

Corollary: If u is drawn from a stationary source with entropy H, then

$$EH(u)=H$$

where E denotes expectation. This corollary follows from the ergodic decomposition of discrete stationary sources [3], [12]. It should be noted, however, that $EH(u)$ might exist for nonstationary sources for which H does not always exist, and can therefore be considered as a generalization of the classical entropy.

It should also be pointed out that sequences exist that can be described by simple algorithms, yet are incompressible by any finite-state machine since they are characterized by $H(u)=h(u)=\log_2\alpha$. For example, let u be the infinite sequence obtained by concatenating the α-ry representations of the natural numbers $1,2,3,\cdots$. That is, for $\alpha=2$, $u=11011100101110111\cdots$. Clearly $h(u)=\log_2\alpha$. It is shown in Theorem 8 of Section III that $H(u)=\log_2\alpha=1$ as well. Hence this sequence is "complex" in the sense of this paper. However, its normalized Kolmogoroff–Solomanoff–Chaitin program-size complexity [4], [5] is zero. The reason for the difference is that we have restricted our encoders and decoders to be finite-state machines. The quantity $h(u)$ (for Case I) or $H(u)$ (for Case II) can be called "the finite-state normalized complexity" of an infinite sequence.

Theorem 6 in Section III states that a modified version of the complexity defined in [1] is a lower bound on $H(u)$, i.e.,

$$c(u) \triangleq \limsup_{n\to\infty} \frac{1}{\left(\frac{n}{\log_2 n}\right)} C(u_1^n) \leqslant H(u) \leqslant h(u).$$

Furthermore, if u is drawn from an ergodic source with entropy H, it follows that (Theorem 7 of Section III):

$$P[|c(u)-H|>\epsilon]=0.$$

Hence $H(u)=c(u)=H$ almost surely.

Finally, a block coding version of a universal sequential data compression algorithm that has been introduced recently [2] is discussed in Section III. This encoding algorithm is asymptotically optimal and its instrumentation complexity grows linearly with the block length.

III. Derivations and Proofs

Lemma 1: The limit $h(u)=\lim_{l\to\infty}h_l(u)$ exists.

Proof: Let $\boldsymbol{h}(u)=\lim_{l\to\infty}\inf h_l(u)$. Hence, for any given arbitrary $\epsilon>0$, a number $q=q(\epsilon)$ can be assigned such that $h_q(u) \leqslant \boldsymbol{h}+\epsilon$. Let n be an integer, and let l be such that

$$lq \leqslant n<(l+1)q, \qquad l=1,2,\cdots.$$

Since there are at most α^q possible continuations of any ql-vector to make it into a $ql+q$ vector, we have

$$2^{nh_n(u)} \leqslant 2^{lqh_{lq}(u)}\alpha^q.$$

But

$$2^{lqh_{lq}(u)} \leqslant (2^{qh_q(u)})^l=2^{lqh_q(u)}.$$

Hence

$$nh_n(u) \leqslant lqh_q(u)+q\log_2\alpha,$$

$$h_n(u) \leqslant \frac{1}{n}lq(\boldsymbol{h}+\epsilon)+\frac{q}{n}\log_2\alpha,$$

$$\limsup_{n\to\infty} h_n(u) \leqslant \boldsymbol{h}+\epsilon$$

where ϵ is arbitrarily small. Q.E.D.

Lemma 2 (Data Processing Lemma):

$$h(\hat{u}) \leqslant h(v) \leqslant h(u)$$

for any finite-state encoder-decoder pair.

Proof: For any positive integer l

$$2^{lh_l(v)} \leqslant S2^{lh_l(u)}$$

since any l-vector u_i^{i+l-1} corresponds to one out of, at most, S different l-vectors $v_{i-t}^{i+l-1-t}$ for any positive integer i (there are S possible initial states). Thus

$$h_l(v) \leqslant h_l(u)+\frac{\log_2 S}{l},$$

$$h(v)=\lim_{l\to\infty}h_l(v) \leqslant \lim_{l\to\infty} h_l(u)=h(u),$$

and similarly $h(\hat{u}) \leqslant h(v)$. Q.E.D.

Theorem 1 (Converse to Coding Theorem): If $h(u)>\log_2\beta$ then $\hat{u}\neq u$.

Proof: Clearly $h(v) \leqslant \log_2\beta$. Hence by Lemma 2

$$h(\hat{u}) \leqslant h(v) \leqslant \log_2\beta.$$

But if $\hat{u}=u$, it follows that $h(\hat{u})=h(u)$. Hence if $u=\hat{u}$, $h(u) \leqslant \log_2\beta$. Q.E.D.

Theorem 3 (Converse to Coding Theorem): If $H(u) > \log_2\beta$, then for any finite-state encoder-decoder pair $d(u,\hat{u}) > \epsilon$ for some $\epsilon > 0$.

Proof: Let $H(u) = \log_2\beta + \delta$, $\delta > 0$. Then there exists some positive ϵ for which

$$\inf_{w: d(u,w) \leqslant \epsilon} h(w) = H_\epsilon(u) > \log_2\beta$$

But by Lemma 2, $h(\hat{u}) \leqslant h(v) \leqslant \log_2\beta$. Therefore $h(\hat{u}) < H_\epsilon(u)$ and so $d(u,\hat{u}) > \epsilon$. Q.E.D.

Theorem 2 (Coding Theorem): For every $n \geqslant \alpha$ there exists a block encoder and a block decoder

$$v_{i+1}^{i+n} = f(u_{i+1}^{i+n}), \quad \hat{u}_{i+1}^{i+n} = g(v_{i+1}^{i+n}), \qquad i = 0, n, 2n, \cdots,$$

where $f(\cdot)$ and $g(\cdot)$ are independent of i, such that $\hat{u} = u$ for *all* sequences u for which $h(u) \leqslant \log_2\beta - \epsilon_n(u)$ where

$$\lim_{n\to\infty} \epsilon_n(u) = 0.$$

Proof: (By construction [6]) Let l be the largest integer such that $l^2\alpha^l \leqslant n$. Let the first L letters of v_{i+1}^{i+n} be taken to be a "list" of all the *distinct* l-vectors in u_{i+1}^{i+n}. Clearly there are no more than α^l such l-vectors. If k is an integer such that $\beta^k \geqslant \alpha^l > \beta^{k-1}$ then

$$L = k\alpha^l \leqslant \left(\frac{l\log_2\alpha}{\log_2\beta} + 1\right)\alpha^l \leqslant \frac{n}{l}\frac{\log_2\alpha}{\log_2\beta}\left(1 + \frac{1}{l}\right). \quad (6)$$

Now parse u_{i+1}^{i+n} as follows:

$$u_{i+1}^{i+n} = u_{i+1}^{i+l}, u_{i+l+1}^{i+2l}, \cdots, u_{i+(m-1)l+1}^{i+ml}, u_{i+ml+1}^{i+n},$$

where $m = \lfloor n/l \rfloor$. (The length of u_{i+ml+1}^{i+n} is less than l if l does not divide n.) There are at most $n/l + 1$ vectors in the parsed u_{i+1}^{i+n}.

The second part of the codeword v_{i+1}^{i+n} is taken to be a sequence of "addresses." Each vector $u_{i+1}^{i+l}, u_{i+l+1}^{i+2l}, \cdots$ is encoded into a q-vector (of letters drawn from V) that points out the place of that l-vector in the list. In the case where l does not divide n, the last vector u_{i+ml+1}^{i+n} is encoded into the address of the l-vector $u_{i+n-l+1}^{i+n}$ (clearly u_{i+ml+1}^{i+n} is a suffix of $u_{i+n-l+1}^{i+n}$, since $ml \geqslant n - l$).

Thus q should satisfy

$$\beta^{q-1} \leqslant 2^{lh_l(u)} \leqslant \beta^q,$$

or

$$q \leqslant \frac{l}{\log_2\beta}\left[h_l(u) + \frac{\log\beta}{l}\right] \leqslant \frac{l}{\log_2\beta}\left[h_l(u) + \frac{\log_2\alpha}{l}\right]$$

since $\alpha \geqslant \beta$. The resulting length of the list of addresses is

$$\left(\left\lfloor \frac{n}{l} \right\rfloor + 1\right)q. \quad (7)$$

If $N = L + (\lfloor n/l \rfloor + 1)q$ turns out to be less than n, prolong it by adding an $(n-N)$-vector whose letters are all equal to the first letter in V (say 0).

In any case β must be large enough so that

$$N \leqslant n. \quad (8)$$

By (6), (7), and (8)

$$N = L + \left(\left\lfloor \frac{n}{l} \right\rfloor + 1\right)m \leqslant \frac{n}{l}\frac{\log_2\alpha}{\log_2\beta}\left(1 + \frac{1}{l}\right) + \left(\frac{n}{l} + 1\right)\frac{l}{\log_2\beta}\left[h_l(u) + \frac{\log_2\alpha}{l}\right]. \quad (9)$$

Let β be large enough to make the right side of (9) smaller than or equal to n, so that

$$n \geqslant n\left[\frac{1}{l}\frac{\log_2\alpha}{\log_2\beta}\left(1 + \frac{1}{l}\right)\right] + \frac{n\left(1 + \frac{l}{n}\right)}{\log_2\beta}\left[h_l(u) + \frac{\log_2\alpha}{l}\right],$$

$$\log_2\beta \geqslant \left(1 + \frac{l}{n}\right)h_l(u) + \frac{1}{l}\log_2\alpha\left(1 + \frac{1}{l}\right) + \left(1 + \frac{l}{n}\right)\left(\frac{\log_2\alpha}{l}\right).$$

Thus a sufficient condition for error-free encoding and decoding is

$$\log_2\beta \geqslant h(u) + \epsilon_n(u)$$

where

$$\begin{aligned}\epsilon_n(u) &= \left(1 + \frac{l}{n}\right)h_l(u) - h(u) + \frac{1}{l}\log_2\alpha\left(1 + \frac{1}{l}\right) + \left(1 + \frac{l}{n}\right)\frac{1}{l}\log_2\alpha \\ &\leqslant \left(1 + \frac{l}{n}\right)h_l(u) - h(u) + \frac{4\log_2\alpha}{l}.\end{aligned}$$

Since l is the largest integer for which $l^2\alpha^l \leqslant n$

$$\lim_{n\to\infty} \epsilon_n(u) = 0.$$ Q.E.D.

Theorem 4 (Coding Theorem): For any $n \geqslant \alpha$ and $\epsilon > 0$, there exists a block encoder and a block decoder

$$v_{i+1}^{i+n} = f(u_{i+1}^{i+n}), \quad \hat{u}_{i+1}^{i+n} = g(u_{i+1}^{i+k}), \qquad i = 0, n, 2n, \cdots$$

such that $d(u,\hat{u}) \leqslant 2\epsilon$ for all sequences such that

$$H(u) \leqslant \log_2\beta - \delta_n(u,\epsilon) - \epsilon$$

where

$$\lim_{n\to\infty} \delta_n(u,\epsilon) = 0.$$

Proof: Let w be a sequence such that $d(u,w) \leqslant \epsilon^2/2$ and such that for any $l > l_0(u,\epsilon)$

$$h_l(w) < H(u) + \epsilon. \quad (10)$$

By definition of $d(u,v)$, there exists an integer m such that for any $k \geqslant m$,

$$\frac{1}{k}\sum_{j=0}^{k-1}\frac{1}{n}D\left(u_{jn+1}^{(j+1)n}, w_{jn+1}^{(j+1)n}\right) \leqslant \epsilon^2.$$

Let

$$\delta(j) = \begin{cases} 0, & \text{if } \frac{1}{n}D\left(u_{jn+1}^{(j+1)n}, w_{jn+1}^{(j+1)n}\right) \leqslant \epsilon = \frac{1}{\epsilon}\epsilon^2 \\ 1, & \text{otherwise} \end{cases}$$

so that

$$\epsilon^2 \geqslant \frac{1}{k}\sum_{j=0}^{k-1}\frac{1}{n}D\left(u_{jn+1}^{(j+1)n}, w_{jn+1}^{(j+1)n}\right) \geqslant \frac{1}{k}\sum_{j=0}^{k-1}\epsilon\delta(j).$$

Thus dividing by ϵ,

$$\frac{1}{k}\sum_{j=0}^{k-1}\delta(j)\leqslant\epsilon.$$

Therefore

$$\limsup_{k\to\infty}\frac{1}{k}\sum_{j=0}^{k-1}\delta(j)\leqslant\epsilon. \tag{11}$$

Thus by assuming that the block coding scheme will fail to accurately encode those n-blocks for which

$$\frac{1}{n}D\left(u_{jn+1}^{(j+1)n},w_{jn+1}^{(j+1)n}\right)>\epsilon,$$

we contribute to the total density of errors in $\hat{u}$ a factor which is smaller than ϵ.

Consider now n-blocks $u_{jn+1}^{(j+1)n}$ $(j=0,1,2,\cdots)$ for which

$$\frac{1}{n}D\left(u_{jn+1}^{(j+1)n},w_{jn+1}^{(j+1)n}\right)\leqslant\epsilon, \tag{12}$$

and recall that by (10)

$$h_l(w)\leqslant H(u)+\epsilon.$$

Therefore the number of distinct l-vectors in the n-vector $w_{jn+1}^{(j+1)n}$ is not larger than $2^{l[H(u)+\epsilon]}$.

The encoding of $u_{jn+1}^{(j+1)n}$ is done as follows. Among all n-vectors that are at Hamming distance not larger than ϵn from $u_{jn+1}^{(j+1)n}$, select that one, say $\hat{u}_{jn+1}^{(j+1)n}$, with the smallest number of distinct l-vectors. Clearly the number of distinct l-vectors in $\hat{u}_{jn+1}^{(j+1)n}$ is less than or equal to that in $w_{jn+1}^{(j+1)n}$, if the $u_{jn+1}^{(j+1)n}$ satisfy (12). Now apply the coding scheme that was used in the proof of Theorem 2 to the vectors $\hat{u}_{jn+1}^{(j+1)n}$ $(j=0,1,2,\cdots)$. It follows from Theorem 2 and (10) that for any n such that $l^2\alpha^l\leqslant n<(l+1)^2\alpha^{l+1}$ and for every $u_{jn+1}^{(j+1)n}$ that satisfies (12),

$$\frac{1}{n}D\left(u_{jn+1}^{(j+1)n},\hat{u}_{jn+1}^{(j+1)n}\right)\leqslant\epsilon$$

provided $\log_2\beta\geqslant h_l(w)+\delta_l$, where $\lim_{l\to\infty}\delta_l=0$. Accordingly, it suffices to have $\log_2\beta\geqslant H(u)+\epsilon+\delta_l$ where $\lim_{l\to\infty}\delta_l=0$.

On the other hand the relative frequency of n-vectors $u_{jn+1}^{(j+1)n}$ such that $(1/n)D(u_{jn+1}^{(j+1)n},w_{jn+1}^{(j+1)n})>\epsilon$ is bounded by (11). Hence

$$d(u,\hat{u})\leqslant 2\epsilon$$

for all sequences u such that

$$\log_2\beta\geqslant H(u)+\epsilon+\delta_n(u,\epsilon) \tag{13}$$

where $\lim_{n\to\infty}\delta_n(u,\epsilon)=0$. Q.E.D.

Theorem 5: If u is drawn from an ergodic source that is characterized by an entropy H, then

$$P\left[H(u)=H\right]=1.$$

Proof: Consider the set of all n-vectors u_1^n that are emitted by the given ergodic source. By the asymptotic equipartition property (AEP) [7], it follows that for any arbitrary positive ϵ

$$\lim_{n\to\infty}P\left[\left|-\frac{1}{n}\log_2P(u_1^n)-H\right|>\epsilon\right]=0$$

where $P(\cdot)$ denotes probability. Therefore for any arbitrarily small $\epsilon>0$, there exists some integer l such that for any $n\geqslant l$

$$P\left[\left|-\frac{1}{n}\log P(u_1^n)-H\right|>\epsilon\right]\leqslant\epsilon.$$

Thus for $n=l$ there is a set S_1 that includes at most $2^{l(H+\epsilon)}$ elements (which are called *typical* l-sequences) such that

$$P\left[u_1^l\in S_1\right]\geqslant 1-\epsilon.$$

Any l-sequence that does not belong to S_1 is called an *atypical* sequence.

Consider the following l different parsings of u:

$$u=u_1^{k-1},u_k^{l+k-1},u_{l+k}^{2l+k-1},\cdots,u_{jl+k}^{(j+1)l+k-1},\cdots,$$
$$j=0,1,\cdots;\ 1\leqslant k\leqslant l,$$

and let $\delta_k(0)=0$,

$$\delta_k(j)=\begin{cases}0, & \text{if } u_{jl+k}^{(j+1)l+k-1}\in S_1,\\ 1, & \text{otherwise,}\end{cases}$$

for $j=1,2,\cdots$ and $1\leqslant k\leqslant l$. Let

$$\lim_{L\to\infty}\frac{1}{L}\sum_{j=0}^{L-1}\delta_k(j)=\delta_k(u).$$

(This limit exists with probability one since $\delta_k(1),\cdots,\delta_k(i),\cdots$ is a stationary sequence.) Furthermore with probability one

$$\frac{1}{l}\sum_{k=1}^{l}\delta_k(u)=1-P_r\left[S_1\right]\leqslant\epsilon,$$

the equality following from the ergodicity of the source. Thus there exists some m $(1\leqslant m\leqslant l)$ for which

$$\delta_m(u)\leqslant\epsilon.$$

For simplicity assume that $m=1$, and consider

$$u=u_1^l,u_{l+1}^{2l},\cdots,u_{jl+1}^{(j+1)l},\cdots,\qquad j=0,1,\cdots.$$

Replace any $u_{jl+1}^{(j+1)l}$ that is not in S_1 by one of the elements of S_1. Let w denote the resulting infinite sequence. Clearly $d(u,w)\leqslant\epsilon$.

Furthermore the number of distinct l-vectors among $w_1^l,w_{l+1}^{2l},\cdots,w_{jl+1}^{(j+1)l}$ is not more than $2^{l(H+\epsilon)}$. It follows that for any positive integer q

$$2^{qlh_{ql}(w)}\leqslant\left[2^{l(H+\epsilon)}\right]^{q-1}\alpha^l\leqslant 2^{ql\left[(H+\epsilon)+(\log_2\alpha/q)\right]}.$$

Hence

$$h(w)=\lim_{q\to\infty}h_{ql}(w)\leqslant H+\epsilon.$$

Therefore by definition

$$H_\epsilon(u)=\inf_{w:\,d(u,w)\leqslant\epsilon}h(w)\leqslant H+\epsilon$$

and

$$H(u)=\lim_{\epsilon\to\infty}H_\epsilon(u)\leqslant H \tag{14}$$

with probability one.

The number of elements in S_1 is at least $2^{l(H-\epsilon)}$, all of which are almost equiprobable (by the AEP). Therefore it

can be shown that with probability one

$$H(u) \geqslant H. \tag{15}$$

This also follows directly from the Coding Theorem 4 and the classical converse theorem of information theory [7]. Thus by (14) and (15)

$$H = H(u)$$

with probability one. Q.E.D.

The following definition has been proposed for the complexity of a sequence [1].

Consider this rule for parsing u_1^n into distinct phrases.

1) A comma is inserted following u_1.

2) Assume that the ith comma comes after the letter u_{k_i}, $1 \leqslant k_i \leqslant n-1$. The next comma will be inserted after the letter $u_{k_{i+1}}$ where $k_{i+1} = k_i + L_i + 1 \leqslant n$ and L_i is the maximal length of a substring $u_{k_i+1} \cdots u_{k_i+L_i}$ such that there exists an integer (or *pointer*) p_i (where $1 \leqslant p_i < k_i$)for which $u_{p_i} \cdots u_{p_i+L_i-1} = u_{k_i+1} \cdots u_{k_i+L_i}$.

As an example we parse a binary sequence of length 16:

k:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
u_k:	0,	0	0	1,	1	0,	1	0	0,	1	0	0	0,	1	0	1

where there is no comma after the last letter since $u_{14}u_{15}u_{16}$ has appeared previously ($p_5 = 5$).

The number of commas in the parsing of an n-sequence is denoted by $C(u_1^n)$ and is called the *complexity* of u_1^n [1]. Let the *normalized complexity* be defined by

$$c(u_1^n) = \frac{C(u_1^n)}{n/\log_2 n}$$

and let $c(u) = \lim_{n\to\infty} \sup c(u_1^n)$. Then [1, Theorem 2]

$$c(u) \leqslant \log_2 \alpha.$$

By a simple generalization of Theorem 2 in [1], the following lemma can be proved.

Lemma 3: $c(u) \leqslant h_l(u)$ for any $l = 1, 2, \cdots$, and hence $c(u) \leqslant h(u)$.

In fact it follows from the next theorem that $c(u)$ is upper bounded by $H(u)$.

Theorem 6: $c(u) \leqslant H(u)$.

Proof: It follows from the definition of $H(u)$ that for any arbitrary small positive ϵ, there exists a sequence w such that $d(u,w) \leqslant \epsilon^2$ and $h(w) \leqslant H(u) + \epsilon$. Hence there exists an integer l_0 such that for any $l > l_0$

$$h_l(w) \leqslant h(w) + \epsilon \leqslant H(u) + 2\epsilon.$$

Now apply the parsing rule to parse u into distinct phrases

$$u = u_1^{l_1}, u_{l_1+1}^{l_2}, \cdots, u_{l_i+1}^{l_{(i+1)}}, \cdots.$$

Clearly

$$\limsup_{n\to\infty} \frac{1}{n} \sum_{i=0}^{C(u_1^n)} D\left(u_{l_i+1}^{l_{(i+1)}}, w_{l_i+1}^{l_{(i+1)}}\right) = d(u,w) \leqslant \epsilon^2.$$

Let

$$\delta(i) = \begin{cases} 0, & \text{if } D\left(u_{l_i+1}^{l_{(i+1)}}, w_{l_i+1}^{l_{(i+1)}}\right) \leqslant \epsilon(l_{i+1} - l_i), \\ 1, & \text{otherwise,} \end{cases}$$

for $i = 0, 1, \cdots$. Then from the proof of Theorem 4

$$\limsup_{n\to\infty} \frac{1}{n} \sum_{i=0}^{C(u_1^n)} \delta(i)(l_{i+1} - l_i) \leqslant \epsilon.$$

Hence there exists an integer m such that for any $n > m$ the total length of vectors for which $\delta(i) = 1$ is bounded by

$$\sum_{i=0}^{C(u_1^n)} \delta(i)(l_{i+1} - l_i) \leqslant 2\epsilon n, \qquad n \geqslant m.$$

Let C_1 be the total number of commas that are contained in the union of the vectors for which $\delta(i) = 1$. Then from [1, Theorem 2]

$$C_1 \leqslant \frac{2\epsilon n}{(1-\delta_n)\log_2(2\epsilon n)} \log_2 \alpha$$

where $\lim_{n\to\infty} \delta_n = 0$. Delete from u_1^n all the vectors $u_{l_i+1}^{l_{(i+1)}}$ for which $\delta(i) = 1$. All the other vectors in the parsed u_1^n have $\delta(i) = 0$. Hence for every $l > l_0$, the total number of distinct phrases of length l that can be found among the undeleted phrases of u_1^n is upper-bounded by

$$2^{lh_l(w)} \binom{l}{\epsilon l} = 2^{lh_l(w)} 2^{E(\epsilon)l} \leqslant 2^{[h(w)+\epsilon+E(\epsilon)]l} \leqslant 2^{[H(u)+2\epsilon+E(\epsilon)]l}$$

where $2^{E(\epsilon)l}$ is the maximum number of l-vectors that are within Hamming distance ϵl of a given l-vector and $2^{lh_l(w)}$ is the number of l-vectors in w. It is easy to show that

$$E(\epsilon) = -\epsilon \log_2 \epsilon - (1-\epsilon)\log_2(1-\epsilon) + \epsilon \log_2(\alpha - 1) + \delta_l,$$

where $\lim_{l\to\infty} \delta_l = 0$. Let C_2 be the total number of commas in the undeleted portions of u_1^n. By a simple generalization of [1, Theorem 2]

$$C_2 \leqslant \frac{n}{\log_2 n}\left[H(u) + 2\epsilon + E(\epsilon) + \delta_n'\right]$$

where $\lim_{n\to\infty} \delta_n' = 0$. Furthermore the total numbers of commas in u_1^n is bounded by

$$C(u_1^n) \leqslant C_1 + C_2.$$

It follows that $c(u) \leqslant H(u) + 4\epsilon + \mathrm{E}(\epsilon)$. Since ϵ is an arbitrary small positive number $c(u) \leqslant H(u)$. Q.E.D.

The following coding scheme, based on the above parsing algorithm, was proposed in [2], [10]. The encoder sequentially parses u_1^n and generates a concatenation of codewords c_i, $1 \leqslant i \leqslant C(u_1^n)$. The first codeword consists of the β-ary expansion of u_1, the first letter of u_1^n. The ith codeword consists of three parts

$$c_i = c_{i1}, c_{i2}, c_{i3},$$

where c_{i1} is the radix-β representation of the ith pointer and the length $l(c_{i1}) = \log_\beta n$, c_{i2} is the radix-β representation of l_i, the distance from the ith comma to the last letter before the $(i+1)$th comma, and $l(c_{i2}) = \lceil 2\log_\beta l_i + 4 \rceil$ ([8], [9]), and finally c_{i3} is the β-ary expansion of the last letter prior to the $(i+1)$th comma, and $l(c_{i3}) = \log_\beta \alpha$. Thus

$$l(c_i) = \lceil \log_\beta n \rceil + \lceil 2\log_\beta l_i + 4 \rceil + \log_\beta \alpha.$$

The total length of the codeword is

$$\sum_{i=1}^{C(u_1^n)} l(c_i) \leqslant C(u_1^n)\left[\log_\beta n + 5 + \log_\beta \alpha\right] + \sum_{i=1}^{C(u_1^n)} 2\log_\beta l_i.$$

Now

$$\sum_{i=1}^{C(u_1^n)} 2\log l_i = 2C(u_1^n) \sum_{i=1}^{C(u_1^n)} \frac{1}{C(u_1^n)} \log l_i$$

$$\leqslant 2C(u_1^n) \log \left[\sum_{i=1}^{C(u_1^n)} \frac{l_i}{C(u_1^n)} \right]$$

$$\leqslant 2C(u_1^n) \log \frac{n}{C(u_1^n)}.$$

But $C(u_1^n)\log(n/C(u_1^n))$ increases monotonically with $C(u_1^n)$ for $n > e \cdot C(u_1^n)$ and $C(u_1^n) \leqslant n/(1-\epsilon_n)\log_\beta n$ where $\lim_{n\to\infty}\epsilon_n = 0$. Hence

$$\sum_{i=1}^{C(u_1^n)} 2\log_\beta l_i \leqslant \frac{n}{1-\epsilon} \frac{\log_\beta \log_\beta n}{\log_\beta n} = n \cdot \delta_n$$

where $\lim_{n\to\infty}\delta_n = 0$. Thus

$$\sum_{i=1}^{C(u_1^n)} l(c_i) \leqslant C(u_1^n)\log_\beta n + n\delta_n'$$

where $\lim_{n\to\infty} \delta_n' = 0$.

In order to guarantee the existence of an error-free block encoding version of the above variable-length encoding algorithm, it is enough to insure that

$$\sum_{i=1}^{C(u_1^n)} l(c_i) \leqslant C(u_1^n)\log_\beta n + n\delta_n' \leqslant n.$$

Thus a sufficient condition for error-free block encoding is

$$\log_2 \beta \geqslant \frac{C(u_1^n)}{n/\log_2 n} + \delta_n' \log_2 \beta$$

or as $n \to \infty$,

$$\log_2 \beta > c(u).$$

The next theorem therefore follows from Theorems 5 and 6 and the classical converse theorem of information theory [7].

Theorem 7: If u is drawn from an ergodic source with an entropy H, then for any arbitrary $\epsilon > 0$

$$P[|c(u) - H > \epsilon] = 0.$$

Thus by Theorems 6 and 7, $c(u) = H(u) = H$ almost surely if u is the output of an ergodic source. The fact that $c(u_1^n) \to H$ in probability for an ergodic source was first established in [10].

From the proof of Theorem 6 it follows that if, among all n-vectors that are at Hamming distance not larger than $\epsilon^2 n$ from $u_{jn+1}^{(j+1)n}$, the vector $\hat{u}_{jn+1}^{(j+1)n}$ has the smallest number of distinct l-vectors, say $2^{\hat{h}(j)l}$, then

$$\frac{C(u_{jn+1}^{(j+1)n})}{n/\log_2 n} \leqslant \hat{h}_l(j) + \delta_\epsilon + \delta_n$$

where $\lim_{n\to\infty}\delta_n \to 0$ and $\lim_{\epsilon\to 0}\delta_\epsilon = 0$. It follows from the proof of the Coding Theorem 4 that the block encoding version of the proposed sequential encoding algorithm is asymptotically optimal in the sense of Theorems 4 and 2. It has been shown [9] that the instrumentation complexity of this algorithm grows only linearly with the block length n. Furthermore, as with sliding-block codes [11], this code improves progressively as n grows, while the induced changes in its structure are slight.

Theorem 8: Let u be the infinite series consisting of the α-ry expansions of the natural numbers $1,2,3,\cdots$. Then $H(u) = \log_2 \alpha$.

Proof: Consider the segment u_s^t of u where $s = s(l) = 1 + \sum_{i=0}^{l} i\alpha^{i-1}$, $t = t(l) = \sum_{i=0}^{l+1} i\alpha^{i-1}$, and $l = 0,1,2,\cdots$. The segment u_s^t corresponds to the α-ry expansion of the natural numbers $\alpha^l, \alpha^l + 1, \cdots, \alpha^{l+1} - 1$.

There are therefore α^l distinct l-vectors in u_s^t. Let w be a sequence such that $h(w) = \log_2 \alpha - 2\epsilon < \log_2 \alpha$. Then there exists an integer k such that for any $l \geqslant k, h_l(\omega) \leqslant \log_2 \alpha - \epsilon$. Thus the number $2^{lh_l(w_s^t)}$ of distinct l-vectors in the segment w_s^t is not larger than $\alpha^l 2^{-\epsilon l}$ for any $l \geqslant k$. Therefore $2^{lh(w_s^t)}/\alpha^l$ tends to zero as l tends to infinity.

The sequence u_s^t consists of $(\alpha^{l+1} - \alpha^l)$ distinct $(l+1)$-vectors. Let us parse u_s^t and w_s^t into $\alpha^{l+1} - \alpha^l$ successive $(l+1)$-blocks and consider the sequences $\hat{u}_s^t$ and $\hat{w}_s^t$ of $(\alpha^{l+1} - \alpha^l)$ l-blocks that are formed by omitting the first letter in each $(l+1)$-block in the parsed u_s^t and w_s^t, respectively.

Let $P(u(l))$ denote the relative frequency of the l-vector $u(l)$ among the $(\alpha^{l+1} - \alpha^l)$ l-blocks that form $\hat{u}_s^t$, and let $p(w(l)|u(l))$ be the relative frequency of the l-vector $w(l)$ among the $(\alpha^{l+1} - \alpha^l)$ l-blocks that form $\hat{w}_s^t$, given that the corresponding l-block in $\hat{u}_s^t$ that pairs with $w(l)$ is $u(l)$.

By construction, $p(u(l)) = \alpha^{-l}$ and

$$d = \frac{1}{t-s} D(u_s^t, w_s^t)$$

$$\geqslant \frac{l}{l+1} \frac{1}{l} \sum_{w(l)} \sum_{u(l)} p(u(l))p(w(l)|u(l))D(u(l),w(l))$$

$$= \frac{l}{l+1} d'.$$

Let

$$\frac{1}{l} I(u(l),w(l)) \triangleq \frac{1}{l} \sum_{w(l)} \sum_{u(l)} p(u(l))p(w(l)|u(l)) \cdot \log_2 \frac{p(w(l)|u(l))}{\sum_{u(l)} p(w(l)|u(l))p(u(l))}.$$

Since the number of distinct l-vectors in w_s^t is bounded by $2^{[\log_2 \alpha - \epsilon]l}$ then ([7])

$$\log_2 \alpha - \epsilon \geqslant h_l(w_s^t) \geqslant \frac{l}{l} H(w(l)) \geqslant \frac{1}{l} I(u(l),w(l)).$$

But $p(u(\cdot)) = \alpha^{-l}$ represents a probability measure of l-vectors emerging from a memoryless source of α-letters. Thus [7, Eq. (9.5.8)]

$$\log_2 \alpha - \epsilon \geqslant R(d') \geqslant \log_2 \alpha + d' \log_2 d' + (1-d')\log_2(1-d') - d'\log_2(\alpha - 1),$$

so as l tends to infinity

$$\log_2\alpha-\epsilon\geqslant R(d)>\log_2\alpha+d\log_2 d+(1-d)\log_2(1-d)-d\log_2(\alpha-1).$$

Thus

$$\lim_{l\to\infty}\frac{1}{t-s}D\left(u_s^t,w_s^t\right)\geqslant d(\epsilon)>0$$

where $d(\epsilon)$ is the solution of

$$-d\log_2 d-(1-d)\log_2(1-d)+d\log(\alpha-1)=\epsilon.$$

Therefore, since $u=u_{s(1)}^{t(1)},u_{s(2)}^{t(2)},\cdots,u_{s(l)}^{t(l)},\cdots$, it follows that

$$d(u,w)>d(\epsilon). \qquad \text{Q.E.D.}$$

Conclusion

1) The fixed-rate finite-state encoding that is discussed in this paper might be considered to be a special case of variable-rate encoding. However, it should be pointed out that in variable-rate encoding there is usually a buffer at the output of the encoder that converts the variable-rate output into a fixed-rate data. Therefore if we restrict ourselves to finite-memory buffers, the fixed-rate model of this paper is the more appropriate one.

The variable-rate encoding case will be discussed in a forthcoming paper with A. Lempel, where it is demonstrated that the sequential data processing algorithm of [2] which is described in Section III is also asymptotically optimal for the variable-rate case, and that $c(u)$ (Theorem 6) is also a lower bound on the compression ratio that can be achieved by any finite-state variable-rate encoder. Furthermore the sequence that is discussed in Theorem 8 is incompressible by any finite-state encoder, even a variable-rate one.

2) Assume that the sequence u is to be transmitted through a noisy channel, for instance a memoryless channel of capacity C. The natural generalization of the converse Theorem 3 states that if $H(u)>C$, then there exists an $\epsilon>0$ such that $P[d(u,\hat{u}))>\epsilon]>\epsilon$.

3) The case where some distortion between u and $\hat{u}$ can be tolerated should lead to a version of the classical rate-distortion theory and a rate-distortion function $R(d)$ for different fidelity criteria. These topics will be discussed elsewhere.

4) Consider the case where the input sequence is *finite*. By following the proofs of Theorems 1,2,3, and 4 it is possible to state similar converse and coding theorems for a finite individual sequence u_1^m. For example, let the number of distinct l-vectors in u_1^m be $2^{lh_l(u_1^m)}$. Then if $\max_l[h_l(u_1^m)-(1/l)\log S]\geqslant\log_2\beta$, we have $u_1^m\neq\hat{u}_1^m$ for any finite-state encoder and decoder with S or fewer states. This is the equivalent of the converse Theorem 2, and follows directly from the proof of Lemma 1 and Theorem 2.

Acknowledgment

This work is an outgrowth of [1] and [2], and the author's debt to A. Lempel, the coauthor of [1] and [2], is obvious. The author wishes to acknowledge with thanks helpful discussions with H. S. Witsenhausen, L. A. Shepp, D. Slepian, S. P. Lloyd, and A. D. Wyner. In particular, A. D. Wyner contributed significantly to the proof of Theorem 5.

References

[1] A. Lempel and J. Ziv, "On the complexity of an individual sequence," *IEEE Trans. Inform. Theory*, vol. IT-22, pp. 75–81, Jan. 1976.

[2] J. Ziv and A. Lempel, "A universal algorithm for sequential data-compression," *IEEE Trans. Inform. Theory*, vol. IT-23 pp. 337–343, May 1977.

[3] R. M. Gray and L. D. Davisson, "Ergodic decomposition of stationary discrete random processes," *IEEE Trans. Inform. Theory*, vol. IT-20, pp. 625–636, Sept. 1974.

[4] P. Martin-Löf, "The definition of random sequences," *Inform. and Contr.*, vol. 9, pp. 602–619, 1966.

[5] G. Chaitin, "A theory of program size formally identical to information theory," *J. ACM*, vol. 22, pp. 329–340, July 1975.

[6] J. Ziv, "Coding of sources with unknown statistics—Part I, probability of encoding error," *IEEE Trans. Inform. Theory*, vol. IT-18, pp. 384–394, May 1972.

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

[8] P. Elias, "Universal codeword sets and representations of integers," *IEEE Trans. Inform. Theory*, vol. IT-21, pp. 194–203, Mar. 1975.

[9] M. Rodeh, "String matching algorithms and their application to data compression," Ph.D. dissertation supervised by S. Even, Dep. Computer Science, Technion, Haifa, Israel, 1976.

[10] I. Shperling "On the asymptotic complexity of sequences," M. Sc. thesis, Dep. Electrical Engineering, Technion, Haifa, Israel, 1976.

[11] R. M. Gray "Sliding-block source coding," *IEEE Trans. Inform. Theory*, vol. IT-21, no. 4, pp. 357–368, July 1975.

[12] K. Winkelbauer, "On discrete information sources," Trans. of the Third Prague Conference on Inform. Theory, Decision Functions, and Random Processes, 1962. Czechoslovak Academy of Sciences, Prague, 1964, pp. 765–830.

[13] R. M. Gray, D. L. Neuhoff and J. K. Omura, "Process definitions of distortion-rate functions and source coding theorems, " *IEEE Trans. Inform. Theory*, vol. IT-21, no. 5, pp. 524–532, Sept. 1975.