HUFFMAN ENCODING UTILIZING THE KRAFT - MCMLLAN INEQUALITY

A thesis submitted

by

Gerald Shea

to

the Faculty of Pure and Applied Science

of the University of Ottawa

in partial fulfillment of the requirements

for the degree of

Master of Science

in the subject of

Mathematics

1970

© Gerald Shea 1972
ACKNOWLEDGMENTS

I wish to express my appreciation to my supervisor, Professor Edward L. Cohen, for his invaluable guidance, his constant encouragement and his constructive suggestions made during the research work and the writing of this thesis. I would also like to thank Professor S. G. S. Shiva, for his suggestion of the problem contained herein and for his consultations with me.

Thanks are due to the National Research Council of Canada and to the Provincial Government of Ontario for their financial aid.
ABSTRACT

Huffman encoding is studied as a logical consequence of the Shannon code for discrete noiseless channels. By means of the Kraft-McMillan inequality, a new algorithm for Huffman encoding is given. In many cases, the algorithm proves to be considerably shorter than previously known methods. An upper bound is found for the lengths of the individual Huffman code words. In conclusion, the efficiency of Huffman codes is discussed.
# TABLE of CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction</td>
<td>1</td>
</tr>
<tr>
<td>Chapter I Instantaneous Codes and the</td>
<td></td>
</tr>
<tr>
<td>Kraft-McMillan Inequality</td>
<td>1</td>
</tr>
<tr>
<td>Entropy of the Source and Shannon Codes</td>
<td>6</td>
</tr>
<tr>
<td>Huffman Encoding</td>
<td>10</td>
</tr>
<tr>
<td>Chapter II Another Algorithm for Huffman Encoding</td>
<td>19</td>
</tr>
<tr>
<td>Chapter III Examples and Applications of the Algorithm</td>
<td>31</td>
</tr>
<tr>
<td>Chapter IV Entropy and Huffman Encoding</td>
<td>49</td>
</tr>
<tr>
<td>List of Definitions</td>
<td>52</td>
</tr>
<tr>
<td>Bibliography</td>
<td>53</td>
</tr>
<tr>
<td>Errata</td>
<td>56</td>
</tr>
</tbody>
</table>
INTRODUCTION

Information theory has its origins in the engineering problem of transmitting data accurately and efficiently through telegraph lines, telephone cable, radio bands, etc. The data may vary from the thousands of words in the English vocabulary to a small set of numbers like \( \{0, 1, \ldots, 9\} \); however, the transmission system is often limited to a few symbols. Binary symbols such as positive or negative charge and electrical pulse or no pulse are common.

In this paper accuracy is not of concern, since an error-free system is assumed. It is also assumed that the encoder produces a finite positive number \( D \) of distinct signals that can be transmitted and then processed by the decoder. Thus, this paper restricts itself to a discrete, noiseless channel—a system that delivers to the decoder a perfect copy of each distinct symbol produced by the encoder.

One measure of efficiency is the time taken to send and receive each message. Let \( \{\alpha_i\}_{i=1}^K \) be a list of possible messages, \( p_i \) the probability that \( \alpha_i \) will be transmitted at any given moment, and \( n_i \) the number of symbols needed to encode \( \alpha_i \). Then the average length of the code for these messages equals \( \sum_{i=1}^K n_i p_i \). To save transmission time, this average should be as short as possible.

A desirable property of an encoding is that it be instantaneous; that is, each code word can be immediately
recognized and uniquely decoded. D. Huffman [15] gave a procedure for finding a minimum average length instantaneous code for any finite set of messages \( \{\alpha_i\}_{i=1}^K \) with fixed probabilities of transmission \( \{p_i\}_{i=1}^K \). The purpose of this paper is to derive another algorithm for Huffman codes. The method given here both shortens in many cases the number of steps in the procedure and aids to elucidate the nature of Huffman codes.

Chapter I presents commonly accepted definitions and lemmas basic to the study of instantaneous codes. The sources from which these details are drawn are given in square brackets; the numbers therein refer to the bibliography.

Since inductive arguments are frequently employed, no mention of this fact is made in the proofs. The form of the argument will reveal when induction is being used.

Chapter II develops a new algorithm for Huffman encoding. For any fixed set of probabilities \( \{p_i\}_{i=1}^K \), a unique instantaneous code is constructed. The algorithm repeatedly reduces the average length of this code. The process ceases when further reductions would destroy the instantaneous property of the encoding. Theorem 1 states that the algorithm has relatively few steps; Theorem 3 shows that the resultant encoding is Huffman.
Chapter III gives examples of codes derived by applying the algorithm in various forms. Some results that simplify computation are illustrated.

Chapter IV describes a Huffman code with reference to the lengths of the individual code words and in terms of its overall efficiency.
CHAPTER I

INSTANTANEOUS CODES and the KRAFT-McMILLAN INEQUALITY

DEFINITION 1: Let \( Z \) be a source of data and let \( A = \{a_i\}_{i=1}^K \) be the total vocabulary of symbols used by \( Z \). Then each \( a_i \) is said to be a word and \( A \) a word list of \( K \) words.

DEFINITION 2: Let \( T \) be a channel consisting of encoder, transmission line, and decoder. Let \( D \geq 2 \) be the number of symbols acceptable to \( T \). Then the set \( \{0, 1, \ldots, D - 1\} \) is said to be the coding alphabet and an integer \( x \), \( 0 \leq x \leq D - 1 \), a code letter.

DEFINITION 3 [1, p. 46]: A block code, or simply a code, relates each word \( a_i \) of a word list \( A \) into a fixed sequence of code letters. These fixed sequences are called code words and are denoted by \( X_i = (x_{1} x_{2} \ldots x_{n_i}) \); \( n_i \) is the length of the code word \( X_i \).

Hence a code may be denoted by \( \{(a_i, X_i)\}_{i=1}^K \).

DEFINITION 4: A code \( \{(a_i, X_i)\}_{i=1}^K \) is said to be uniquely decodable [1, p. 48] if every finite sequence \( X_1 X_2 \ldots X_\mu = x_{1} x_{2} \ldots x_{\lambda} \) of \( \mu \) code words and \( \lambda \) code letters is distinct from every different finite sequence of \( \nu \) code words \( Y_1 Y_2 \ldots Y_\nu = y_{1} y_{2} \ldots y_{\lambda} \) of \( \lambda \) code letters.

Therefore a uniquely decodable code can decode any code letter sequence \( x_{1} x_{2} \ldots x_{\lambda} \) at most one way.

DEFINITION 5: A code \( \{(a_i, X_i)\}_{i=1}^K \) is said to be instantaneous [1, p. 50] if each word received through a discrete noiseless channel can be immediately and uniquely decoded.
An instantaneous code is a uniquely decodable code with the property that \( x_1 = x_1 x_2 \ldots x_{n_1} \) is recognized as soon as it is received. That is, no investigation of the subsequent code letters is needed to assure the decoder that \( x_1 x_2 \ldots x_{n_1} \) is a complete code word, namely, \( X_1 \).

**DEFINITION 6:** Let \( X_1 = x_1 x_2 \ldots x_{n_1} \) be a code word. The sequence of code letters \( (x_1 x_2 \ldots x_\lambda) \) with \( \lambda < n_1 \) is called a *prefix* of \( X_1 \) [1, p. 50].

**LEMMA 1:** A code \( \{(x_i, X_i)\}_{i=1}^K \) is instantaneous iff no \( X_j \) is a prefix of \( X_i \), for all \( i \) and all \( j \) not equal to \( i \). [1, p. 51]

**Proof:** The sufficiency. Let the sequence

\[
X_1 X_2 \ldots = x_1 x_2 \ldots x_{n_1} y_1 y_2 \ldots y_{n_j} \ldots
\]

be sent through the channel. By the prefix property\( x_1 x_2 \ldots x_{n_1} \) is distinct from all other code words. Also, \( x_1 x_2 \ldots x_m \) is not a code word for \( m < n_1 \) as no prefix of \( X_1 \) is, and hence \( x_1 x_2 \ldots x_m \) will not be decoded. Again \( x_1 x_2 \ldots x_{n_1} y_1 \), \( x_1 x_2 \ldots x_{n_1} y_1 y_2 \), \ldots are not code words as \( X_1 \) is not a prefix of any other code word. Therefore \( X_1 \) is uniquely decipherable as soon as \( x_1 x_2 \ldots x_{n_1} \) is received.

The necessity. Assume there exist code words \( X_i \) and \( X_j \) such that \( X_1 \) is a prefix of \( X_j \). Consider the sequence

\[
X_1 X_2 \ldots = x_1 x_2 \ldots x_{n_1} y_1 y_2 \ldots y_{n_j} \ldots
\]
If $n_i = n_j$, $X_i = X_j$, and the sequence cannot be uniquely decoded. Assume $n_i < n_j$. Then $x_1 x_2 \ldots x_{n_i}$ could be either $X_i$ or the first $n_i$ code letters of $X_j$. No decision could be made, if at all, until at least one more code letter is examined. In any case the sequence cannot be both uniquely and immediately decoded. QED

**Lemma 2:** [17, 20, 22; 1, pp. 53 - 61]

**The Kraft - McMillan Inequality**

Let $\{n_i\}_{i=1}^K$ be a set of positive integers. Then there exists a uniquely decodable code $\{(a_i, X_i)\}_{i=1}^K$ with these word lengths if, and only if,

$$\sum_{i=1}^K D^{-n_i} \leq 1.$$ 

**Proof:** The sufficiency. Let $\{n_i\}_{i=1}^K$ be a set of positive integers such that $\sum_{i=1}^K D^{-n_i} \leq 1$. It is enough to show that an instantaneous code with these word lengths can be constructed.

Let $N = \max \{n_i\}_{i=1}^K$, and let $q_j$, $j = 1, 2, \ldots, N$ be the number of $n_i$ equal to $j$. That is, $\sum_{j=1}^N q_j = K$.

Then $\sum_{j=1}^N q_j D^{-j} \leq 1$ or $\sum_{j=1}^N q_j D^{N-j} \leq D^N$. Hence

$$q_N \leq D^N - q_1 D^{N-1} - q_2 D^{N-2} - \ldots - q_{N-1} D.$$ 

Since $q_j \geq 0$, $j = 1, 2, \ldots, N$ and $D > 0$, the following sequence of inequalities is formed by successively dividing by $D$ and rearranging the terms.

$$q_{N-1} \leq D^{N-1} - q_1 D^{N-2} - q_2 D^{N-3} - \ldots - q_{N-2} D$$

$$\ldots \ldots \ldots \ldots \ldots \ldots$$
\[ q_2 \leq D^2 - q_1 D \]

\[ q_1 \leq D \]

By means of Lemma 1, an instantaneous code is constructed as follows:

Since \( q_1 \leq D \), \( q_1 \) distinct code words of length equal to 1 are formed from \( \{0, 1, 2, \ldots, q_1 - 1\} \). \( D - q_1 \) distinct prefixes, namely \( \{q_1, q_1 + 1, \ldots, D - 1\} \), can be used for longer code words. Thus \( (D - q_1)D \) code words of two code letters may be formed with distinct prefixes.

As \( q_2 \leq (D - q_1)D \), the necessary code words are constructed.

Let \( Y_j = D^j - q_1 D^{j-1} - q_2 D^{j-2} - \ldots - q_{j-1} D \), for some \( j \), \( 1 \leq j \leq N \). Assume \( q_j \) code words of length \( j \) have been formed satisfying the prefix property. As \( q_j \leq Y_j \), \( Y_j - q_j \) distinct prefixes of length \( j \) remain. In fact, \( (Y_j - q_j)D \) code words of length \( j + 1 \) may be formed under the prefix property. As \( q_{j+1} \leq (Y_j - q_j)D \), these code words can be constructed.

The necessity. Let \( I = \{(a_i, X_i)\}_{i=1}^K \) be a uniquely decodable code with code word lengths \( \{n_i\}_{i=1}^K \). That is, each finite sequence of \( r \) code letters can be decoded at most one way. Let \( Q_r \) be the number of sequences of \( q \) code words that can be formed so that each sequence has exactly \( r \) code letters. Since each sequence must be distinct, \( Q_r \leq D^r \), as there are only \( D^r \) distinct \( r \)-tuples.
Consider \( (\sum_{i=1}^{K} D^{-n_i})q = (D^{-n_1} + D^{-n_2} + \ldots + D^{-n_K})q \).

Let \( N = \max \{n_i\}_{i=1}^{K} \). Since each of the \( K^q \) terms of the expansion is of the form \( D^{-n_1}D^{-n_2} \ldots D^{-n_q} \) and there are \( q^r \) terms such that \( n_1 + n_2 + \ldots + n_q = r \), implying that \( q \leq r \leq qN \), then

\[
(\sum_{i=1}^{K} D^{-n_i})q = \sum_{r=q}^{qN} q^r D^{-r} \leq \sum_{r=q}^{qN} q^r D^{-r} \leq qN - q + 1 \leq qN .
\]

Since \( N \) is constant, and the inequality holds for all \( q \),
\[
\sum_{i=1}^{K} D^{-n_i} \leq 1 .
\]

QED

By means of the prefix property, given the \( \{n_i\}_{i=1}^{K} \) of an instantaneous code, a representation of the code may be easily constructed [1, pp. 52 - 53].

One method is to construct "trees": Let \( D = 2 \).

Note \( 2^{-1} + 2^{-3} + 2^{-3} + 2^{-4} + 2^{-4} \leq 1 \). Procede as follows.

```
  0
 / \   \
0  1
 / \   \
0  1
 / \   \
0  1
```

Then

\[
\begin{align*}
n_1 &= 1 : : X_1 = 0 \\
n_2 &= 3 : : X_2 = 1 0 0 \\
n_3 &= 3 : : X_3 = 1 0 1 \\
n_4 &= 4 : : X_4 = 1 1 0 0 \\
n_5 &= 4 : : X_5 = 1 1 0 1
\end{align*}
\]
DEFINITION 7: Let $I = \{(a_i, x_i)\}_{i=1}^K$ be a code with code word lengths $\{n_i\}_{i=1}^K$. Define

$$S_I = \sum_{i=1}^K D_i^{-n_i}.$$ 

Hence by the prefix property and Definition 7 with the Kraft-McMillan Inequality, an instantaneous code may be redefined by $I = \{(a_i, n_i)\}_{i=1}^K$ such that $S_I \leq 1$. Then $\{(a_i, x_i)\}_{i=1}^K$ may be constructed as illustrated on the previous page.

**ENTROPY of the SOURCE and SHANNON CODES**

DEFINITION 8: Let $Z$ be a discrete source and $\{a_i\}_{i=1}^K$ the word list of all possible words used by $Z$. Let $\{p_i\}_{i=1}^K$ be given such that $p_i$ is the probability that $a_i$ is emitted ($i = 1, 2, \ldots, K$) with $\sum_{i=1}^K p_i = 1$. If successive words $a_1 a_2 \ldots$ are statistically independent, $Z$ is said to be a zero-memory source [1, p. 14].

A zero-memory source is completely described by $\{a_i\}_{i=1}^K$ and $\{p_i\}_{i=1}^K$ and shall be denoted $Z = \{(a_i, p_i)\}_{i=1}^K$.

For the statistical independence of $Z$ gives the probability of any sequence of words $a_1 a_2 \ldots a_n$ as $p_1 p_2 \ldots p_n$.

DEFINITION 9: The entropy $H_D(Z)$ [1, p. 14] of a zero-memory source $Z = \{(a_i, p_i)\}_{i=1}^K$ with respect to a code with $D$ code letters is defined by

$$H_D(Z) = -\sum_{i=1}^K p_i \log_D p_i.$$
Shannon [30] indicates that \(-\log_D p_i\) serves to measure the uncertainty of the event \(a_i\). Hence it gives the value of the information that \(a_i\) did occur. It also justifies the use of approximately \(-\log_D p_i\) code letters to encode \(a_i\).

**DEFINITION 10:** Let \(Z = \{(a_i, p_i)\}_{i=1}^K\) be a zero-memory source. The \(n\)-th extension \(Z^n\) of \(Z\) [1, p. 20] is defined by \(Z^n = \{(\beta_j, \pi_j)\}_{j=1}^K\) where the \(\beta_j\) are distinct sequences of code words \(\beta_j = (a_{i_1} a_{i_2} \ldots a_{i_n})\) for \(i = 1, 2, \ldots, K^n\).

Therefore \(\pi_j = p_{i_1} p_{i_2} \ldots p_{i_n}\). Also

\[
\prod_{j=1}^{K^n} \pi_j = \prod_{i=1}^K p_{i_1} \prod_{i=2}^K p_{i_2} \ldots \prod_{i=n}^K p_{i_n} = 1.
\]

Thus \(Z^n\) is a zero-memory source.

\(Z^n\) is also called a "finite delay source". In practice, the coder waits until a sequence \(a_{i_1} a_{i_2} \ldots a_{i_n}\) of \(n\) words is received from \(Z\) and then encodes this sequence as a single word from the zero-memory source \(Z^n\).

**LEMMA 3:** Let \(Z = \{(a_i, p_i)\}_{i=1}^K\) be a zero-memory source and \(Z^n = \{(\beta_j, \pi_j)\}_{j=1}^K\) its \(n\)-th extension. Then

\[
H_D(Z^n) = nH_D(Z).
\]

**Proof [1, p. 21]:** Let each \(\beta_j = (a_{i_1} a_{i_2} \ldots a_{i_n})\). Then each \(\pi_j = p_{i_1} p_{i_2} \ldots p_{i_n}\), \(j = 1, 2, \ldots, K^n\). Thus
\[ H_D(Z^n) = -\sum_{j=1}^{K^n} \pi_j \log_D \pi_j \]
\[ = -\sum_{j=1}^{K^n} \pi_j \log_D p_{i_1} - \sum_{j=1}^{K^n} \pi_j \log_D p_{i_2} - \ldots - \sum_{j=1}^{K^n} \pi_j \log_D p_{i_n} \]

For any \( r, 1 \leq r \leq n \),
\[ -\sum_{j=1}^{K^n} \pi_j \log_D p_{i_r} = -\sum_{i=1}^{K} p_{i_r} \log_D p_{i_r} \sum_{i=1}^{K} p_{i_1} \ldots \sum_{i=1}^{K} p_{i_n} \]
\[ = -\sum_{i=1}^{K} p_{i_r} \log_D p_{i_r} = H_D(Z). \]

Therefore \( H_D(Z^n) = nH_D(Z) \). QED

**DEFINITION 11:** Let \( Z = \{(a_1, p_1)\}_{i=1}^{K} \) be a zero-memory source and \( I = \{(a_1, n_1)\}_{i=1}^{K} \) an instantaneous code on \( Z \).
The average length of \( I \), denoted by \( L_I \), equals \( \sum_{i=1}^{K} n_i p_i \).

[1, p. 66].

For a source \( Z = \{(a_1, p_1)\}_{i=1}^{K} \) and an instantaneous code \( I = \{(a_1, n_1)\}_{i=1}^{K} \) on \( Z \), \( H_D(Z) = -\sum_{i=1}^{K} p_i \log_D p_i \),
\( S_I = \sum_{i=1}^{K} n_i \) and \( L_I = \sum_{i=1}^{K} n_i p_i \) are independent of the word list \( \{a_i\}_{i=1}^{K} \). Henceforth the interests of this paper centre on the probabilities \( \{p_i\}_{i=1}^{K} \) characterizing the source \( Z \) and the code word lengths \( \{n_i\}_{i=1}^{K} \) of \( I \).

For this purpose, the code \( I \) shall be denoted by the relation \( \{(p_1, n_1)\}_{i=1}^{K} \) with the set of probabilities \( \{p_i\}_{i=1}^{K} \) fixed. To return to the \( \{(a_1, n_1)\}_{i=1}^{K} \) form of \( I \), let \( Z = \{(a_1, p_1)\}_{i=1}^{K} \). Then \( a_i = p_i + n_i \) \((i = 1, 2, \ldots, K)\) gives the customary form, aside from trivial differences in the case of equiprobable words.
DEFINITION 12: Let \( \{p_i\}_{i=1}^K \) be a fixed set of probabilities. Then the Shannon code [30; I, p. 72] \( E = \{ (p_i, n_i) \}_{i=1}^K \) is the unique code defined by:

\[-\log_D p_i \leq n_i < 1 - \log_D p_i \quad (i = 1, 2, \ldots, K) .\]

By the left-hand inequality, \( D^{-n_i} \leq p_i \quad (i = 1, 2, \ldots, K) .\)

Therefore \( S_i = \sum_{i=1}^K D^{-n_i} \leq \sum_{i=1}^K p_i = 1 .\) Hence, \( E \) is an instantaneous code.

LEMMA 4 [30; I, pp. 72-73]:

SHANNON'S THEOREM ON NOISELESS CODING

Let \( Z = \{ (\alpha_i, p_i) \}_{i=1}^K \) be a zero-memory source and \( Z^n = \{ (\beta_j, \pi_j) \}_{j=1}^N \) be the \( n \)-th extension. Let \( E \) and \( E^n \) be the Shannon codes \( \{ (p_i, n_i) \}_{i=1}^K \) and \( \{ (\pi_j, \nu_j) \}_{j=1}^N \) for \( Z \) and \( Z^n \), respectively. Let \( L_n \) be the average length of \( E^n \). Then

\[ H_D(Z) \leq L_n / n < H_D(Z) + 1/n .\]

Proof: By Definition 12, \( E^n = \{ (\nu_j, \pi_j) \}_{j=1}^N \) is defined by:

\[-\log_D \pi_j \leq \nu_j < 1 - \log_D \pi_j \quad (j = 1, 2, \ldots, K^n) .\]

Then

\[-\sum_{j=1}^{K^n} \pi_j \log_D \pi_j \leq \sum_{j=1}^{K^n} \pi_j \nu_j < \sum_{j=1}^{K^n} \pi_j = \sum_{j=1}^{K^n} \pi_j \log_D \pi_j .\]

That is, \( H_D(Z^n) \leq L_n < 1 + H_D(Z^n) .\) By Lemma 3, \( H_D(Z^n) = nH_D(Z) .\) Hence \( H_D(Z) \leq L_n / n < H_D(Z) + 1/n .\) QED

\( L_n \) is the average length of the code words in \( E^n \) per word \( \beta_j = (\alpha_{i_1}, \alpha_{i_2}, \ldots, \alpha_{i_n}) \) in \( Z^n \). Hence, \( L_n / n \) is the average length of \( E^n \) per \( \alpha_i \) emitted by \( Z \).
Therefore by increasing the complexity of the encoder and decoder to transmit $K^n$ words, the average length of the code per word from $Z$ can be made arbitrarily close to the entropy of the source.

However Shannon's code, while instantaneous, is often inefficient. In the next section, an encoding is introduced which is, in a sense, the best instantaneous code.

**Huffman Encoding**

**Definition 13:** Let $\{p_i\}_{i=1}^K$ be a fixed set of probabilities. A Huffman (or optimum) code $[15]$ $C = \{(p_i, \tilde{r}_i)\}_{i=1}^K$ is an instantaneous code of minimum average length $L_C$.

Huffman's procedure for constructing an optimum code for a fixed set of probabilities $\{p_i\}_{i=1}^K$ is in two parts:

**Part I:**

(I : 1) List all probabilities such that $p_1 \geq p_2 \geq \ldots \geq p_K$.

(I : 2) Let $A \equiv K \pmod{D-1}$ such that $2 \leq A \leq D$.

(I : 3) Define a new probability list $\{p'_i\}_{i=1}^{K'}$ such that

\[
p'_i = p_i, \quad i = 1, 2, \ldots, K - A
\]

\[
p'_{K'} = \sum_{i=K-A+1}^K p_i. \quad \text{Hence } K' = K - A + 1.
\]

Repeat steps (I: 1, 2, 3) for $\{p'_i\}_{i=1}^{K'}$ until a list is formed with $D$ or fewer probabilities.

**Lemma 5:** In the second (or subsequent) list, let $K'$ be the number of probabilities. Then $K' \equiv D \pmod{D-1}$; so $A \equiv K' \pmod{D-1}$, $2 \leq A \leq D$, implies $A = D$.

**Proof:** Let $A \equiv K \pmod{D-1}$. Then by (I : 3) $A$ of the probabilities are summed to one new probability in the second list and $K' = K - A + 1 \equiv 1 \equiv D \pmod{D-1}$. 
Next let $K'$ be the number of probabilities in any list after the first and assume $K' \equiv D \pmod{D-1}$. Then for this list $A = D$. Therefore the succeeding list has $K'' = K' - D + 1 \equiv 1 \equiv D \pmod{D-1}$.

QED

There are $\left[\frac{(K-A)}{(D-1)}\right] + 1$ probability lists in this form of encoding. Let $K_i, j = 0, 1, \ldots, (K-A)/(D-1)$ represent the number of probabilities in each list, starting with the shortest list. Denote these lists by $P_j = \{p_i^j\}_{i=1}^{K_j}$, $j = 0, 1, \ldots, (K-A)/(D-1)$. [The $j$ of $p_i^j$ is a superscript.]

By Lemma 5 and step (I : 3) $K_0 = D$. Thus $K_j = D + j(D-1)$, $j = 0, 1, \ldots, \left[\frac{(K-A)}{(D-1)}\right] - 1$. For $j' = \frac{(K-A)}{(D-1)}$, $K_{j'} = K$.

Using this terminology, (I : 3) becomes:

For $P_{j+1}$, $(K-A)/(D-1) > j \geq 1$, define a new probability list $P_j = \{p_i^j\}_{i=1}^{K_j}$ such that:

$$p_i^j = p_{i+1}^{j+1}, \ i = 1, 2, \ldots, K_j - A = K_j - 1$$

$$p_j^j = \sum_{i=K_j}^{K_j + 1} p_i^{j+1}.$$

Part II:

(II : 1) For $j = 0$, assign the code words 0, 1, ..., $(D-1)$ to $p_1^0, p_2^0, \ldots, p_D^0$ respectively.

(II : 2) For $1 \leq j < \frac{(K-A)}{(D-1)}$, let $p_\alpha^j = \sum_{i=K_j}^{K_j + 1} p_i^{j+1}$ as in (I : 3), $1 \leq \alpha \leq K_j$. Denote the code word for $p_i^j$ of length $n_i^j$ by $X_i^j$, $i = 1, 2, \ldots, K_j$. Then assign code words for $P_{j+1}$ as follows: $p_i^{j+1}$ is encoded by $X_i^j$ for $i = 1, 2, \ldots, \alpha - 1$ and by $X_{i+\alpha}^j$ for $i = \alpha, \alpha + 1, \ldots, K_j - \alpha$. 
The last $D$ probabilities of $P_{j+1}$, that is

$$p_{k_{j+1}}^{j+1}, p_{k_{j+2}}^{j+1}, \ldots, p_{k_{j+l}}^{j+1}$$

are encoded by adding the code letters $0, 1, \ldots, (D-1)$ to the prefix $X_j^j$, respectively.

Hence the code word lengths $n_i^{j+1}$, $i = 1, 2, \ldots, K_j - 1$, are the same as the $n_i^j$, $i = 1, 2, \ldots, \alpha - 1, \alpha + 1, \ldots, K_j + 1$.

But $n_1^{j+1} = n_\alpha^j + 1$, $i = K_j, K_j + 1, \ldots, K_j + 1$.

**EXAMPLE:** For $D = 4$ and $K = 14 \equiv 2 \pmod{3}$.

$$P_4 : C_4 \quad P_3 : C_3 \quad P_2 : C_2 \quad P_1 : C_1 \quad P_0 : C_0$$

$$\begin{array}{cccccc}
.20 & 1 & .20 & 1 & .20 & 1 \rightarrow .42 \ 0 \\
.18 & 3 & .18 & 3 & .18 & 3 \rightarrow .20 \ 2 \\
.10 & 01 & .10 & 01 & .10 & 01 \rightarrow .12 \ 00 \\
.10 & 02 & .10 & 02 & .10 & 02 \rightarrow .12 \ 00 \\
.10 & 03 & .10 & 03 & .10 & 03 \rightarrow .18 \ 3j \\
.06 & 20 & .06 & 20 & .06 & 20 \rightarrow .06 \ 20 \\
.06 & 21 & .06 & 21 & .06 & 21 \rightarrow .06 \ 21 \\
.04 & 22 & .04 & 22 & .04 & 22 \rightarrow .04 \ 22 \\
.04 & 23 & .04 & 23 & .04 & 23 \rightarrow .04 \ 23 \\
.04 & 000 & .04 & 000 \\
.03 & 001 & .03 & 001 \\
.02 & 003 & .02 & 003 \\
.02 & 0020 \\
.01 & 0021
\end{array}$$

$$S_{C_4} = 1, \ i = 0, 1, 2, 3 \text{ and } S_{C_4} = 254 / 256. \text{ Also } L_{C_4} = 1.77.$$
In showing that the final encoding is Huffman, it is seen that each code \( C_j, j = 0, 1, \ldots, (K-A)/(D-1) \) is instantaneous and of minimum average length; that is, all are Huffman codes.

**LEMMA 6:** Let \( C_j = \{(p_{i}^{j}, n_{i}^{j})\}_{i=1}^{K} \), \( j = 0, 1, \ldots, (K-A)/(D-1) \) be the encoding associated with the probability list \( P_j \).

Then \( S_{C_j} = 1, j = 0, 1, \ldots, [(K-A)/(D-1)] - 1 \). If \( K \equiv D \pmod{D-1} \), then for \( j' = (K-A)/(D-1) \), \( S_{C_j'} = 1 \);

if not, \( S_{C_j'} < 1 \).

**Proof:** Let \( j = 0 \). Then \( n_{i}^{0} = 1, i = 1, 2, \ldots, D \) and \( S_{C_0} = D^{D-1} = 1 \).

Assume for some \( j, 1 \leq j < [(K-A)/(D-1)] - 1 \), that \( S_{C_j} = 1 \). Then, by Lemma 5 \( K_j \equiv K_{j+1} \equiv D \pmod{D-1} \) and \( K_{j+1} = K_j + D - 1 \) by (I : 3). Let \( p_{i}^{j} \) denote some probability in \( P_j \) such that \( p_{i}^{j} = \sum_{i=1}^{K_j} p_{i}^{j+1} \). Then encode \( P_{j+1} \) such that \( n_{i}^{j+1} = n_{i}^{j} + 1, i = K_j, K_j + 1, \ldots, K_{j+1} \). Thus

\[
S_{C_{j+1}} = \sum_{i=1}^{K_{j+1}} D^{-n_{i}^{j+1}} = \sum_{i=1}^{K_{j}} D^{-n_{i}^{j}} + \sum_{i=K_{j}+1}^{K_{j+1}} D^{-n_{i}^{j}+1}
\]

\[
= \sum_{i=1}^{K_{j}} D^{-n_{i}^{j}} + D^{-(n_{i}^{j}+1)} = 1, since S_{C_j} = \sum_{i=1}^{K_{j}} D^{-n_{i}^{j}} = 1.
\]

If \( K \equiv D \pmod{D-1} \), the above holds for \( C_{j'}, j' = (K-A)/(D-1) \). If \( A \equiv K \pmod{D-1} \) is less than \( D \),

\[
\sum_{i=1}^{K_{j'}} D^{-n_{i}^{j'}} < D^{-(n_{i}^{j}+1)} \text{ implying that } S_{C_{j'}} < 1.
\]
LEMMA 7: The encoding \( C = \{(p_i^j, n_i^j)\}_{i=1}^{K_j} \) assigned to \( P_j \), 
\( j = 1, 2, \ldots, (K-A)/(D-1) \) by (II : 2) is a Huffman code.
Proof: For all \( j = 1, 2, \ldots, (K-A)/(D-1) \), \( S_C \leq 1 \),
by Lemma 6; so, \( C_j \) is instantaneous by Lemma 2—Huffman
encoding is a "tree" method of of assigning code words with
the prefix property. Hence it is sufficient to show that
\( L_C \) is minimum, \( j = 1, 2, \ldots, (K-A)/(D-1) \).

For \( j = 0 \), \( C_0 = \{(p_i^0, n_i^0)\}_{i=1}^{D} \) has \( n_i^0 = 1 \),
\( i = 1, 2, \ldots, D \). Thus, no \( n_i^0 \) can be reduced, and
\( L_{C_0} = \sum_{i=1}^{D} n_i^0 p_i^0 = \sum_{i=1}^{D} p_i^0 = 1 \) is minimum.

Assume for some \( C_j \), \( 1 \leq j < (K-A)/(D-1) \), that \( L_{C_j} \)
is minimum. Let \( p_i^j \) be a probability in \( P_j \) such that
\( p_i^j = \frac{K_j+1}{i=1} p_i^{j+1} \). Then \( n_i^{j+1} > n_i^j + 1 \), \( i = K_j, K_j+1, \ldots, K_j+1 \).
To see this, assume \( n_i^j < n_i^j \), \( K_j < \beta < K_j+1 \). Thus
\( S_{C_j+1} = \sum_{i=1}^{K_j-1} D_i^j + D_i^\beta + \sum_{i=K_j+1}^{K_j+1} D_i^j \)
\( = \sum_{i=1}^{K_j} D_i^j + \sum_{i=K_j+1}^{K_j+1} D_i^j > 1 \), since \( K_j+1 > K_j + 2 \).

But this contradicts Lemma 6.

It is enough if \( n_i^{j+1} = n_i^j + 1 \), \( i = K_j, K_j+1, \ldots, K_j+1 \).
Then \( L_{C_j+1} = \sum_{i=1}^{K_j+1} p_i^{j+1} n_i^{j+1} = \sum_{i=1}^{K_j} p_i^{j+1} n_i^{j+1} + \sum_{i=K_j+1}^{K_j+1} p_i^{j+1} n_i^{j+1} \)
\( = \sum_{i=1}^{K_j} p_i^{j+1} n_i^j + \sum_{i=K_j+1}^{K_j+1} p_i^{j+1} = L_{C_j} + \sum_{i=K_j+1}^{K_j+1} p_i^{j+1} \).
Since \( \{ p_i^{j+1} \}_{i=K_j}^{K_j+1} \) is the set of lowest probabilities in \( P_{j+1} \) by (I : 1) and (I : 3), and \( L_{C_j} \) is minimum, then \( L_{C_{j+1}} \) is minimum.

QED

Huffman [15] pointed out that the code word length \( n_i \) of the encoding equals the number of times \( p_i \) was summed to form a larger probability and these new probabilities were in turn combined. Neumann [23, 9] gave a simple algorithm for finding how often these combinations were performed. An example taking the probability list previously encoded with \( D = 4 \) and \( K = 14 \equiv 2 \pmod{3} \) follows on the next page.

Neumann's method is actually a shorthand for noting in every list how each probability was formed. Just as in Huffman's procedure, first \( A \equiv K \pmod{D-1} \), \( 2 \leq A \leq D \), probabilities are summed and then \( D \) more are successively combined. This continues until the sum 1.00 is reached. But with each summation the construction of the new probability from the original probabilities is described in the second column of table 2.

With respect to the example, the first step sums the two lowest probabilities .02 and .01 to .03, and denotes this by \( .03 \ | \ *2 \) in table 2, step 2. In the second step the four lowest probabilities are summed, namely .04, .03, and .02 from table 1 and .03 from table 2. But since \( .03 \ | \ *2 \) represents an implied summation of two probabilities from the original list, this implication is maintained by shifting \*2 one place to the right. Thus \( .12 \ | \ *3*2 \) notes that .12 is the summation of five probabilities, three
<table>
<thead>
<tr>
<th>Step</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>P_4</td>
<td>.28</td>
<td>.18</td>
<td>.10</td>
<td>.10</td>
<td>.10</td>
<td>.06</td>
</tr>
<tr>
<td>P_3</td>
<td>.20</td>
<td>.18</td>
<td>.10</td>
<td>.10</td>
<td>.10</td>
<td>.06</td>
</tr>
<tr>
<td>P_2</td>
<td>.20</td>
<td>.18</td>
<td>.10</td>
<td>.10</td>
<td>.10</td>
<td>(.06)</td>
</tr>
<tr>
<td>P_1</td>
<td>.20</td>
<td>.18</td>
<td>(.10)</td>
<td>(.10)</td>
<td>(.10)</td>
<td>(.10)</td>
</tr>
<tr>
<td>P_0</td>
<td>(.20)</td>
<td>(.18)</td>
<td>( .10)</td>
<td>(.10)</td>
<td>(.10)</td>
<td>(.10)</td>
</tr>
<tr>
<td>P_0</td>
<td>( .10)</td>
<td>( .10)</td>
<td>( .10)</td>
<td>( .10)</td>
<td>( .10)</td>
<td>( .10)</td>
</tr>
</tbody>
</table>

**Note:** The table entries are numerical values with some values in parentheses.
of which have been combined once and two twice. Note that in step 5, \( .20 \mid ^4 \) and \( .42 \mid ^3 \) are summed by first adding the corresponding summation indicators, shifting this total one place to the right and then registering the number of probabilities summed from table 1.

Continuing this way, eventually the \( K \) probabilities are summed to 1.00. Then the implied summations may be used to assign Huffman code word lengths \( \hat{n}_i \). There will be two \( \hat{n}_1 = 1 \), seven \( \hat{n}_1 = 2 \), three \( \hat{n}_1 = 3 \), and two \( \hat{n}_1 = 4 \). Since \( p_1 \geq p_2 \geq \ldots \geq p_K \), then \( \hat{n}_1 \leq \hat{n}_2 \leq \ldots \leq \hat{n}_K \), and so \( \{ (p_i, \hat{n}_i) \}_{i=1}^{K} \) is formed. Finally by the prefix property, an encoding such as the one previously illustrated can be constructed.

Neumann's method thus has one more step then Huffman's, but the last step merely reflects the assigning of code words of length \( n_1^0 = 1 \) to the probability list \( P_0 \) in Huffman's procedure.

However, both Huffman's and Neumann's procedure have at least \( (K-A)/(D-1) \) steps. As \( K \) increases, this becomes more and more unwieldy. In the next section another algorithm will be discussed, based on a method suggested by Shiva and Sheng[32]. There, two properties of the probability set \( \{ p_i \}_{i=1}^{K} \) will be assumed.

First, let \( K \equiv D \pmod{D-1} \) or equivalently \( K \equiv 1 \pmod{D-1} \). For if \( K \equiv A \pmod{D-1} \), \( 2 \leq A < D \), then substitute

\[
P_{K'} = \sum_{i=K-A+1}^{K} p_i
\]

as in (I : 3) of the Huffman encoding. Thus \( K' \equiv D \pmod{D-1} \) as desired (Lemma 5). Having found
the Huffman encoding \( \{(p_i, \bar{n}_i)\}_{i=1}^{K'} \), by Lemma 7 the last step is to assign \( \bar{n}_i = \bar{n}_a + 1 \), \( i = K-A+1, K-A+2, \ldots, K \) as in (II : 2).

Note that in the binary case with \( D = 2 \), \( K \equiv 1 \pmod{D-1} \) automatically.

Secondly, without loss of generality, take \( \{p_i\}_{i=1}^{K} \) such that \( p_1 \geq p_2 \geq \cdots \geq p_K \). Hence in a Huffman encoding \( \{(p_i, \bar{n}_i)\}_{i=1}^{K}, \bar{n}_1 \leq \bar{n}_2 \leq \cdots \leq \bar{n}_K \).
CHAPTER II

ANOTHER ALGORITHM for HUFFMAN ENCODING

DEFINITION 14: Let \( I = \{(p_i, n_i)\}^K_{i=1} \) be a code. Then define
\[
R_I = 1 - S_I = 1 - \sum_{i=1}^{K} D^{-n_i}.
\]

LEMMA 8: Let \( E = \{(p_i, n_i)\}^K_{i=1} \) be a Shannon code. Then
\[
R_E = \sum_{\mu=1}^{n_K} a_\mu (D - 1)D^{-\mu}; \text{ for integers } a_\mu, 0 \leq a_\mu < D.
\]

Proof: Since a Shannon code is instantaneous, \( S_E \leq 1 \), giving
\[
R_E \geq 0. \text{ Define } S_1 = \sum_{i=1}^{K} D^{n_K-n_i}, S_2 = D^{n_K}. \text{ That is,}
\]
\[
S_E = S_1 / S_2. \text{ Now } D \geq 2 \text{ implies that for any integer } a \geq 0,
\]
\[
D^a - 1 = 0 \text{ or } D^a - 1 = (D - 1)(D^{a-1} + D^{a-2} + \ldots + 1).
\]
Thus \( (D - 1)|(D^{n_K-n_i} - 1) \), \( i = 1, 2, \ldots, K \). Hence
\[
S_1 \equiv K \equiv 1 \pmod{D-1} \text{ and } S_2 \equiv 1 \pmod{D-1}.
\]
Therefore \( S_2 - S_1 \equiv 0 \pmod{D-1} \) and
\[
R_E = (S_2 - S_1) / S_2 = B(D - 1) / D^{n_K}
\]
for some integer \( B \geq 0 \). By repeated application of the Euclidean algorithm,
\[
B = \sum_{\mu=1}^{n_K} a_\mu D^{n_K-\mu}, \text{ for integers } a_\mu, 0 \leq a_\mu < D.
\]
Finally
\[
R_E = (D - 1)D^{-n_K}(\sum_{\mu=0}^{n_K} a_\mu D^{-\mu}) = \sum_{\mu=1}^{n_K} a_\mu (D - 1)D^{-\mu}.
\]
QED
DEFINITION 15: Let \( \{p_i\}_{i=1}^K \) be a fixed set of probabilities. An instantaneous code \( M = \{(p_i, n_i^0)\}_{i=1}^K \) is said to be \textit{minimal} if \( L_M \) is a minimum such that each \( n_i^0 \) is less than or equal to \( n_i \) of the Shannon code \( E \).

By definition, a minimal code is not necessarily a Huffman code, and vice versa. However, Theorem 3 will show the two codes to be equivalent.

\textbf{Lemma 9:} For any fixed set of probabilities \( \{p_i\}_{i=1}^K \), there exists a minimal code \( M = \{(p_i, n_i^0)\}_{i=1}^K \). Also \( S_M = 1 \).

\textbf{Proof:} Let \( E = \{(p_i, n_i)\}_{i=1}^K \) be the Shannon code for \( \{p_i\}_{i=1}^K \). Then there are at most \( \prod_{i=1}^K n_i \) distinct codes with each code word length less than or equal to \( n_i \). Since the Shannon code is instantaneous, instantaneous codes form a non-empty subclass of this set of codes. Hence \( M = \{(p_i, n_i^0)\}_{i=1}^K \) can be chosen from the finite subclass such that \( L_M \) is minimum.

By Lemma 2, \( S_M \leq 1 \). Assume \( S_M < 1 \). Define \( S_1 = \sum_{i=1}^K n_i^0 - n_i^0 \) and \( S_2 = D_K n_i^0 \). Then \( S_M = S_1 / S_2 \) and \( S_1 < S_2 \). As in the proof of Lemma 8, \( S_1 \equiv S_2 \equiv 1 \pmod{D-1} \). For \( 0 \leq b \leq D - 2 \), \( S_1 + b \neq S_2 \pmod{D-1} \) giving \( S_1 + b < S_2 \). Thus \( S_1 + D - 1 \leq S_2 \). And therefore \( S' = S_M + (D - 1)D^{-n_K} = (S_1 + D - 1) / S_2 \leq 1 \).

So, there exists a code \( M' = \{(p_i, n_i')\}_{i=1}^K \) with \( n_i' = n_i^0 \), \( i = 1, 2, \ldots, K-1 \), \( n_K' = n_K^0 - 1 \) and \( S_M' = S' \leq 1 \).

By Lemma 2, \( M' \) is instantaneous and since \( p_K' > 0 \), \( L_M' < L_M \). This contradicts the minimality of \( M \). \( \text{QED} \)
DEFINITION 16: Let \( E = \{(p_i, n_i)\}_{i=1}^K \) be a Shannon code. Define
\[
v_i = p_i^{D-n_i}, \quad i = 1, 2, \ldots, K.
\]
Then \( v_i \) is said to be the valuation of \( n_i \).

THEOREM 1: For a fixed set of probabilities \( \{p_i\}_{i=1}^K \), a minimal code \( M = \{(p_i, n_i)\}_{i=1}^K \) can be found in at most \([\log_D p^-K]\) steps, where \( p^-K \) is the lowest probability.

Before considering the algorithm needed to prove Theorem 1, some machinery must be introduced.

DEFINITION 17: The codes considered in the algorithm are called the \( \mu \)-th outer codes, \( \mu = n^-K, n^-K-1, \ldots, 2, 1 \) and are denoted \( E_\mu = \{(v_i, p_i, n_i)\}_{i=1}^K \). \( E_{n^-K} \), the original outer code, is composed of \( E = \{(p_i, n_i)\}_{i=1}^K \), the Shannon code of \( \{p_i\}_{i=1}^K \), plus the valuations \( v_i \) of \( E \).
\( E_\mu \), \( \mu = n^-K-1, n^-K-2, \ldots, 2, 1 \) represent codes evolved from \( E_{n^-K} \) at successive stages of the algorithm. They are said to be reduced outer codes.

LEMMA 10: Let \( \mu', n^-K \geq \mu \geq 1 \), represent any distinct stage of the algorithm and let \( E_\mu \) be the \( \mu \)-th outer code. Then for integers \( a_\nu \), \( R_{E_\mu} = \sum_{\nu=1}^{\mu} a_\nu (D - 1)^{D-\nu} \), \( 0 \leq a_\nu < D \).

Proof: For \( \mu = n^-K \), this is immediate from Lemma 8.

Assume the hypothesis for some \( \mu, n^-K \geq \mu \geq 1 \).
Let \( \mu' = \mu - 1 \). Then
\[
R_{E_\mu} = \sum_{\nu=1}^{\mu'} a_\nu (D - 1)^{D-\nu} + a_\mu (D - 1)^{D-\mu}.
\]
Let $M$ be the minimal code for $\{p_i\}_{i=1}^K$ of Theorem 1. Then by Lemma 9, $R_M = 0$. Also by Lemma 8, $0 \leq a_\mu < D$.

Hence $a_\mu (D - 1)D^{-\mu}$ can be subtracted from $R_{E_\mu}$ by, and only by, reducing sufficient $n_i \geq \mu$. As all such $n_i$ are listed in $E_\mu$, they are then reduced, giving, for $0 \leq a_\nu < D$

$$R_{E_\mu} = R_{E_\mu} - a_\mu (D - 1)D^{-\mu} = \sum_{\nu=1}^{\mu'} a_\nu (D - 1)D^{-\nu}.$$ QED

**LEMMA 11:** Let $\mu$, $n_K \geq \mu \geq 1$, be a distinct stage of the algorithm and $E_\mu = \{(v_i, p_i, n_i)\}_{i=1}^K$ the $\mu$-th outer code. Then each $n_i \geq \mu$, $1 \leq i \leq K$, is in a set of the form

$$\{n_i \geq \mu : j = 1, 2, \ldots, F\}$$

such that

$$\sum_{j=1}^F n_i^j D^{-\mu} = D^{-\mu}.$$  

If $m$ is the number of such sets in $E_\mu$, then $D \mid (m - a_\mu)$.

**Proof:** First let $\mu = n_K$. Then as $n_i \leq n_K$ ($i = 1, 2, \ldots, K$), $n_i \geq \mu = n_K$ implies that $n_i = \mu$. Thus each $n_i \geq \mu$ is in a singleton set $\{n_i = \mu\}$ and $D^{-1} = D^{-\mu}$. Let $m$ be the number of the sets $\{n_i = n_K\}$ in $E_{n_K}$.

(*) Let $b_\nu$, $\nu = 1, 2, \ldots, \mu - 1$ be the number of sets $\{n_i = \nu\}$ and let $b_\mu = m$. Then

$$S_{E_\mu} = \sum_{i=1}^K -n_i = \sum_{\nu=1}^\mu b_\nu D^{-\nu}.$$  

By the Euclidean algorithm, $b_\mu = qD + b$, $0 \leq b < D$ and $b, q \geq 0$ integers. Now

$$b_\mu D^{-\mu} = (qD + b)D^{-\mu} = qD^{1-\mu} + bD^{-\mu}.$$  


Therefore \( D^{1-\mu} | (S_{E_{\mu}} - bD^{-\mu}) \). Also by Lemma 10,

\[
R_{E_{\mu}} = \sum_{\nu=1}^{\mu} a_{\nu} (D - 1)^{D^{-\nu}} \text{ so that } D^{1-\mu} | (R_{E_{\mu}} - a_{\mu} (D - 1)^{D^{-\mu}}).
\]

Since \( \mu \geq 1 \), \( R_{E_{\mu}} + S_{E_{\mu}} = 1 \) implies \( D^{1-\mu} | (R_{E_{\mu}} + S_{E_{\mu}}) \).

Thus \( D^{1-\mu} | \{[b + a_{\mu} (D - 1)]D^{-\mu}\} \). Then \( D | [b + a_{\mu} (D - 1)] \)
or \( D | (b - a_{\mu}) \). As \( 0 \leq b, a_{\mu} < D \), \( b = a_{\mu} \). Now \( m = b_{\mu} = qD + b = qD + a_{\mu} \) and so \( D | (m - a_{\mu}) \).

Secondly, assume the hypothesis holds for some arbitrary \( \mu \), \( n_{K} \geq \mu \geq 1 \). Let \( \mu' = \mu - 1 \). Since \( m \geq 0 \), and \( D | (m - a_{\mu}) \), there are at least \( a_{\mu} \) sets \{\( n_{i_{j}} \geq \mu' : j = 1, 2, \ldots, F \) with \( l_{j=1}^{F} D^{-i_{j}} = D^{-\mu} \). In these \( a_{\mu} \) sets, the \( n_{i_{j}} \) are replaced by \( n'_{i_{j}} = n_{i_{j}} - 1 \). These sets are then in the form

\[
\{n'_{i_{j}} \geq \mu' : j = 1, 2, \ldots, F\} \text{ and } l_{j=1}^{F} D^{-i_{j}} = l_{j=1}^{F} D^{1-n'_{i_{j}}} = D^{-\mu'}.
\]

By the hypothesis, \( D | (m - a_{\mu}) \). Therefore the remaining \( (m - a_{\mu}) \) sets are combined \( D \) at a time giving \( (m - a_{\mu}) / D \) sets of the form \{\( n'_{i_{j}} \geq \mu > \mu' : j = 1, 2, \ldots, F' \)\} where \( n'_{i_{j}} \)
is unchanged from \( n_{i_{j}} \) and \( l_{j=1}^{F'} D^{-i_{j}} = l_{j=1}^{F'} D^{1-n'_{i_{j}}} = D^{-\mu'} \).

Finally any sets \( \{n_{i_{1}} = \mu'\} \) unreduced from the original
outer code \( E_{n_{K}} \) have \( D^{-i_{1}} = D^{-\mu'} \). Let \( m \) now be the number of sets \( \{n'_{i_{j}} \geq \mu' : j = 1, 2, \ldots, F'\} \). By the previous argument
\((*)\), \( D | (m - a_{\mu}) \).

QED
DEFINITION 18: Let \( \mu, n_K \geq \mu \geq 1 \) be a distinct stage of the algorithm with \( \mu \)-th outer code \( E_\mu \). Let \( m_\mu \) be the number of sets in \( E_\mu \) of the form given in Lemma 10. Define \( N_\mu, \lambda = \{ n_{ij} \geq \mu : j = 1,2, \ldots, E_\lambda \} \) such that
\[
\sum_{j=1}^{E_\lambda} D^{-n_{ij}} = D^{-\mu} ; \quad \lambda = 1, 2, \ldots, m_\mu .
\]
These \( N_\mu, \lambda \) are said to be the \( \mu \)-blocks of \( E_\mu \).

Thus each \( \mu \)-block may have one element \( n_{i1} = \mu \) or several \( n_{ij} > \mu, j = 1, 2, \ldots, E_\lambda \).

DEFINITION 19: Let \( E_\mu, n_K \geq \mu \geq 1 \) be a \( \mu \)-th outer code with \( \mu \)-blocks \( N_\mu, \lambda, \lambda = 1, 2, \ldots, m_\mu \). Let \( E_\mu' \) be a reduced outer code such that \( \mu' = \mu - 1 \) with \( \mu' \)-blocks \( N_\mu', \lambda', \lambda' = 1, 2, \ldots, m_\mu' \). \( V_\lambda \), the common valuation of \( N_\mu, \lambda \) is defined inductively as follows:

For every singleton \( \mu \)- or \( \mu' \)-block, say \( N_\mu, \lambda = \{ n_{i1} = \mu \} \),
\[
V_\lambda = v_{i1} = p_{i1}D \quad (\text{hence for } \mu = n_K, V_\lambda, \lambda = 1, 2, \ldots, m_\mu \text{ is well defined as each } N_\mu, \lambda \text{ is a singleton set}).
\]

For the reduced outer code \( E_\mu' \), \( a_\mu \) common valuations \( V_\lambda' \), are found by dividing \( V_\lambda \) by \( D \), and \( (m_\mu - a_\mu) / D \) of the \( V_\lambda \), are derived by averaging the \( V_\lambda \), taking \( D \) of them at a time.

To be specific,
\[
V_\lambda' = V_\lambda / D, \quad \lambda = \lambda' = 1, 2, \ldots, a_\mu .
\]
Let \( A_\lambda, = \{ \lambda : a_\mu + D(\lambda' - a_\mu - 1) + 1 \leq \lambda \leq a_\mu + D(\lambda' - a_\mu) \} \),
\[
\lambda' = a_\mu + 1, a_\mu + 2, \ldots, a_\mu + (m_\mu - a_\mu) / D .
\]
Then \( V_\lambda' = \sum_{A_\lambda,} V_\lambda / D \).

The remaining \( \mu' \)-blocks are singleton sets \( \{ n_{i1} = \mu' \} \) unreduced from \( E_{n_K} \) with \( V_\lambda' = v_{i1} \).
For programming convenience, each \( n_{ij} \) in \( N_{\mu, \lambda} \) is given the valuation \( V_{ij} = V_{\lambda} \).

ALGORITHM: Repeat steps (i) and (ii) for \( \mu = n_K, n_K-1, \ldots \) successively until step (iii) can be applied.

(i) List all \( (v_1, p_1, n_1) \) so that \( v_1 > v_2 > \cdots > v_K \).

(ii) If for the first listed \( n_g \), say \( g = 1, 2, \ldots, G \)

\[
\sum_{g=1}^{G-1} D^{-n_g} < R_{E_{\mu}} < \sum_{g=1}^{G} D^{-n_g},
\]

a reduced outer code is formed as follows—otherwise proceed to step (iii):

Consider the \( \mu \)-blocks \( N_{\mu, \lambda} = \{ n_{ij} \geq \mu : j = 1, 2, \ldots, F_{\lambda} \} \), \( \lambda = 1, 2, \ldots, m_\mu \). If \( a_\mu \neq 0 \), replace each \( n_{ij} \) in \( N_{\mu, \lambda} \) by \( n_{ij}' = n_{ij} - 1 \), \( \lambda = 1, 2, \ldots, a_\mu \). Hence there are new common valuations \( V_{\lambda'} = V_{\lambda} / D \) and

\[
\sum_{j=1}^{F_{\lambda'}} D i_{j} = D^{1-\mu}, \quad \lambda = \lambda', l = 1, 2, \ldots, a_\mu.
\]

Now for any value of \( a_\mu \), \( D \mid (m_\mu - a_\mu) \) by Lemma 11. Combine the \( \mu \)-blocks \( N_{\mu, \lambda}, \lambda = a_\mu+1, a_\mu+2, \ldots, m_\mu \) taking them \( D \) at a time and averaging their common valuations. That is, \( N_{\mu, \lambda}, N_{\mu, \lambda+1}, \ldots, N_{\mu, \lambda+D-1} \) (where \( \lambda = a_\mu+1, a_\mu+D+1, a_\mu+2D+1, \ldots, m_\mu-\mu \)) are replaced by \( N_{\mu, \lambda'} = \{ n_{ij} \geq \mu : j = 1, 2, \ldots, F_{\lambda'} \} \) such that \( \sum_{j=1}^{F_{\lambda'}} D i_{j} = D^{1-\mu} \) and \( a_\mu+1 \leq \lambda' \leq a_\mu + (m_\mu - a_\mu) / D \). Here the \( n_{ij}' \) remain unchanged from the \( n_{ij}, \) \( F_{\lambda'} = F_{\lambda} + F_{\lambda+1} + \ldots + F_{\lambda+D-1} \)

and \( V_{\lambda'} = (V_{\lambda} + V_{\lambda+1} + \ldots + V_{\lambda+D-1}) / D \).
Let $\mu' = \mu - 1$. There is a reduced outer code $E_{\mu}$, with $\mu'$-blocks $N_{\mu',\lambda'}$, common valuations $V_{\lambda'}$, and

\[ R_{E_{\mu}} = R_{E_{\mu}} - a_{\mu} D^{-\mu} = \sum_{\nu=1}^{\mu'} a_{\nu} D^{-\nu}. \]

Return to step (i).

(iii) Here, for the first listed $n_g$, $g = 1, 2, \ldots, G$,

\[ \sum_{g=1}^{G} n_g = R_{E_{\mu}}. \]

A final reduced outer code $M$ is given by replacing the $n_g$ by $n_g - 1$, $g = 1, 2, \ldots, G$.

As $R_M = 0$, no further reductions to an instantaneous code are possible (Lemma 2).

**END OF ALGORITHM**

LEMMA 12: Let $E_{\mu}$, $n_K \geq \mu \geq 1$, be a $\mu$-th outer code.

Then for each $\mu$-block $N_{\mu,\lambda} = \{ n_{i_j} \geq \mu : j = 1, 2, \ldots, F_{\lambda} \}$ with common valuation $V_{\lambda}$, $\lambda = 1, 2, \ldots, m_{\mu}$

\[ \sum_{j=1}^{F_{\lambda}} p_{i_j} = V_{\lambda} D^{-\mu}. \]

**Proof:** For $\mu = n_K$, this is immediate from Definitions 17 and 19. Assume the lemma holds for some $\mu$, $n_K \geq \mu \geq 1$, and let $\mu' = \mu - 1$. There are three cases to be considered in the formation of $\mu'$-blocks and common valuations in $E_{\mu'}$:

(a) If $a_{\mu} \neq 0$, for $\lambda' = \lambda = 1, 2, \ldots, a_{\mu}$ the $\mu'$-blocks are constructed such that $N_{\mu',\lambda'} = \{ n_{i_j} \geq \mu' : j = 1, 2, \ldots, F_{\lambda} \}$ with $n_{i_j} = n_{i_j} - 1$, for all $n_{i_j}$ in $N_{\mu,\lambda}$. Hence

\[ V_{\lambda'} D^{-\mu'} = (V_{\lambda} / D)(D^{1-\mu}) = V_{\lambda} D^{-\mu} = \sum_{j=1}^{F_{\lambda}} p_{i_j} = \sum_{j=1}^{F_{\lambda'}} p_{i_j}. \]

(b) For a fixed $\lambda'$, $a_{\mu'} + 1 \leq \lambda' \leq a_{\mu} + (m_{\mu} - a_{\mu}) / D$,

let $A_{\lambda'}$, again be defined by:

\[ A_{\lambda'} = \{ \lambda : a_{\mu} + D(\lambda' - a_{\mu} - 1) + 1 \leq \lambda \leq a_{\mu} + D(\lambda' - a_{\mu}) \}. \]
Then $N_{\mu^{'}, \lambda^{'}} = \{n_{ij}^{'}, \geq \mu : j = 1, 2, \ldots, F_{\lambda^{'}}\}$ such that

$$F_{\lambda^{'}} = \sum_{A_{\lambda^{'}}} F_{\lambda}, \quad \sum_{j=1}^{F_{\lambda^{'}}} D^{-n_{ij}^{'}} = \sum_{A_{\lambda^{'}}} F_{\lambda} - n_{ij}^{'}, \quad D \cdot D^{-\mu} = D^{-\mu^{'}},$$

and $V_{\lambda^{'}} = \sum_{A_{\lambda^{'}}} V_{\lambda} / D$. Then

$$D^{-\mu^{'}}V_{\lambda^{'}} = (D^{1-\mu})(\sum_{A_{\lambda^{'}}} V_{\lambda} / D) = D^{-\mu} \sum_{A_{\lambda^{'}}} V_{\lambda} = \sum_{A_{\lambda^{'}}} \sum_{j=1}^{F_{\lambda^{'}}} p_{ij} = \sum_{j=1}^{F_{\lambda^{'}}} p_{ij}.$$

(c) For $\lambda^{'}, a_{\mu} + (m_{\mu} - a_{\mu}) / D, \quad N_{\mu^{'}, \lambda} = \{n_{ij}, \mu^{'})$

and by Definitions 17 and 19 $p_{ij} = v_{ij}D^{-\mu^{'}} = V_{\lambda^{'}}D^{-\mu^{'}}$. QED

Since, for all $n_{ij}$ in $N_{\mu, \lambda}$, $v_{ij}$ is set equal to $V_{\lambda}$, then $V_{\lambda}D^{-\mu} = v_{ij} \sum_{j=1}^{F_{\lambda}} D^{-n_{ij}}, \quad j = 1, 2, \ldots, F_{\lambda}$. Consequently $V_{\lambda}D^{-\mu} = \sum_{j=1}^{F_{\lambda}} v_{ij}D^{-n_{ij}}$. By Definition 17, for all $n_{i} < \mu, v_{i} = p_{i}D^{-n_{i}}$ or $p_{i} = v_{i}D^{-n_{i}}$.

**DEFINITION 20:** Let $E$ be the original outer code. Let $M$ be the final reduced outer code and $m_{o}$ the final value of $\mu$ in step (iii) of the algorithm. The change in the average length of $E_{\mu}$, the $\mu$-th outer code, is denoted by $\Delta L_{\mu}$ and is defined as:

$$\Delta L_{\mu} = L_{E_{\mu}} - L_{E_{\mu-1}}; \quad \mu = n_{K}, n_{K} - 1, \ldots, m_{o} + 1$$

$$\Delta L_{m_{o}} = L_{E_{m_{o}}} - L_{M}.$$
DEFINITION 21: The change in the average length, denoted by $\Delta L$, is given by

$$\Delta L = \sum_{\mu=\mu_0}^{n_K} \Delta L_{\mu}.$$ 

Proof of Theorem 1: For the original outer code $E = E^{n_K}_{\mu}$, $n_K \leq \lceil -\log_D p_K \rceil + 1$, where $p_K$ is the lowest non-zero probability and $[x]$ is the integer part of $x$. Since $\mu_0 \geq 1$, the number of reduced outer codes in the algorithm $(n_K - \mu_0) \leq \lceil -\log_D p_K \rceil$.

Since for a fixed set $\{p_i\}_{i=1}^K$ the original outer code $E$ is unique and $\Delta L = L_E - L_M$, to show $M$ is minimal it is sufficient that $\Delta L$ be maximum.

For $\mu = n_K, n_K - 1, \ldots, \mu_0 + 1, a_{\mu} = 0$ implies that $\Delta L_{\mu} = 0$. If $a_{\mu} \neq 0$, by Lemma 12,

$$\Delta L_{\mu} = \sum_{\lambda=1}^{a_{\mu}} \sum_{j=1}^{r_j} p_{ij}^0 = \sum_{\lambda=1}^{a_{\mu}} V_\lambda D^{-\mu}$$

Since $V_\lambda, \lambda = 1, 2, \ldots, a_{\mu}$, is maximum by step (i) of the algorithm, $\Delta L_{\mu}$ is maximum.

For $\mu_0$, there exists some $g$, and $g = 1, 2, \ldots, G$ such that $\sum_{g=1}^{G} D^{-n_g} = R_{E_{\mu_0}}$.

Hence $\Delta L_{\mu_0} = \sum_{g=1}^{G} p_g = \sum_{g=1}^{G} v_g D^{-n_g} \geq v_{G+1} \sum_{g=1}^{G} D^{-n_g} = v_{G+1} R_{E_{\mu_0}}$.

By step (i), $v_g \geq v_1, i \geq G + 1$. Also $v_{G+1} \geq v_1/D \geq v_g/D$, $g = 2, 3, \ldots, G$. For assume $v_{G+1} < v_1/D$. Since $n_1 < 1 - \log_D p_1$, $v_1 < D$ and so $v_{G+1} < 1$. Thus all $n_i^0$ in $M$ have been reduced at least once from the $n_i$ in $E$. 


Then \( R_E \geq (D - 1) \sum_{i=1}^{K} n_i^1 - n_1^1 = (D - 1)S_E \). That is,
\[
R_E + S_E \geq D > 1 .
\]
But by Definition 14, this is impossible. Therefore all alternative sets \( \{n_i^h : h = 1, 2, \ldots, H\} \)
\[
\sum_{h=1}^{H} n_i^h = R_{E_{\mu_0}}
\]
such that \( v_{i,h} \leq v_{G} \) have valuations \( v_{i,h} \leq v_{G} \), so \( \Delta L_{\mu_0} \) is maximum. By Definition 21, \( \Delta L \) is maximum. Since each code word of \( M \) has length bounded by the \( n_i \) of the Shannon code \( E \), \( M \) is a minimal code. QED

**THEOREM 2:** Let \( E = \{ (p_i, n_i^1) \}_{i=1}^{K} \) be the Shannon code for a fixed set of probabilities \( \{p_i^1\}_{i=1}^{K} \). Let
\[
A = \{ (p_i, n_i^1) : n_i^1 > n_i \}_{i=1}^{K} .
\]
Then the code \( M = \{ (p_i, n_i^o) \}_{i=1}^{K} \) of minimum average length such that \( n_i^o < n_i^1 \), \( i = 1, 2, \ldots, K \) is a minimal code with \( n_i^o \leq n_i \leq n_i^1 \).

**Proof:** By Theorem 1, this is trivial when \( A = E \). So assume for some \( j \), \( 1 \leq j \leq K \), that \( n_j^1 > n_j \). Since \( n_i^1 \geq n_i \), \( i = 1, 2, \ldots, K \), then \( S_A < S_E \leq 1 \). By Definition 12, \( n_j < 1 - \log D p_j \leq n_j^1 \) implying that
\[
v_j = p_j^1 D \leq v_j^1 = p_j D \leq v_j^1 .
\]
Hence \( n_j^1 \) is shortened to \( n_j \) with valuation always greater than or equal to \( D \). That is, \( A \) is reduced to \( E \). By the algorithm, \( E \) is reduced to \( M \). As the valuation has always been maximum, the proof of Theorem 1 shows \( M \) to be a minimal code. QED

**THEOREM 3:** Let \( E = \{ (p_i, n_i^1) \}_{i=1}^{K} \) be the Shannon code for a fixed set of probabilities \( \{p_i^1\}_{i=1}^{K} \). Let
\[
C = \{ (p_i, \tilde{n}_i) \}_{i=1}^{K}
\]
be a Huffman encoding of \( \{p_i^1\}_{i=1}^{K} \). Then for all \( i \), \( \tilde{n}_i \leq n_i \). That is, every minimal code is a Huffman code.
Proof: Assume for some \( j, \ 1 \leq j \leq K, \) that \( \tilde{n}_j > n_j. \)
Let \( C' = \{(p_i, n'_i) : n'_i = \max[n_i, \tilde{n}_i] ; \}_{i=1}^K. \) By this definition, \( C' \) can be returned to \( C \) by reducing the \( n'_i \) of the form \( n'_i = \frac{n_i}{q} > \frac{\tilde{n}_i}{q} \) (i.e. \( v'_i < D \)). By Theorem 2, \( C' \) can also be reduced to a minimal code \( M. \)
Now there exists an \( n'_j = \tilde{n}_j > n_j, \) such that \( v'_j \geq D. \)
Since \( n'_i > n_i, \ i = 1, 2, \ldots, K, \) then \( S_{C'} < S_E < l; \)
so \( (D - 1)D^{-n'_j} \leq S_E - S_{C'} \leq S_M - S_{C'} = S_C - S_{C'} \) as \( S_M = S_C = 1 \) by Lemmas 6 and 9.

Therefore \( C' \) can be reduced to either \( C^o \) or \( M^o \) where \( S_{M^o} = S_{C^o} = S_{C'}, \) and \( (D - 1)D^{-n'_j} \leq 1, \) and \( L_{M^o} = L_{C'}, -p_j, \ L_{C^o} = L_{C'}, -p_q \) \( (p_q = \sum_{q=1}^Q p_{i_q} \) is defined to be the maximum possible decrease in \( L_{C'} \) by reducing \( n'_i = \frac{\tilde{n}_i}{q} > n_i \) and \( D^{-n'_j} = \sum_{q=1}^Q D^{-\tilde{n}'_j}. \) But \( v'_j \geq D > v_{i_q}, \)
\( q = 1, 2, \ldots, Q, \) hence \( p_{i_q} \sum_{q=1}^Q p_{i_q} < p_j \sum_{q=1}^Q \) or \( p_{i_q} < p_j \).
Then \( \sum_{q=1}^Q p_{i_q} < p_j \sum_{q=1}^Q p_{i_q} - n'_j. \)
That is, \( L_{M^o} < L_{C^o}. \) By Theorem 2, the minimal code \( M \)
is found such that \( L_{M^o} - L_M \geq L_{C^o} - L_{C}. \) Therefore
\( 0 < L_{C^o} - L_{M^o} \leq L_{C} - L_{M}. \) Hence \( L_M < L_{C}, \) contradicting \( L_C \) is minimum. \( \text{QED} \)
CHAPTER III

EXAMPLES and APPLICATIONS of the ALGORITHM

The algorithm as presented in Chapter II is in a form suitable for computer programming. Examples will be given, some of which show methods that can simplify computation.

EXAMPLE 1: For \( D = 2 \). The original encoding \( \{(p_i, \tilde{r}_i)^K_{i=1}\} \) is taken from Huffman [15]. Twelve steps were needed in his procedure.

<table>
<thead>
<tr>
<th>( p_i )</th>
<th>( n_i )</th>
<th>( v_i )</th>
<th>( \tilde{r}_i )</th>
</tr>
</thead>
<tbody>
<tr>
<td>.20</td>
<td>3</td>
<td>1.60</td>
<td>2</td>
</tr>
<tr>
<td>.18</td>
<td>3</td>
<td>1.44</td>
<td>3</td>
</tr>
<tr>
<td>.10</td>
<td>4</td>
<td>1.60</td>
<td>3</td>
</tr>
<tr>
<td>.10</td>
<td>4</td>
<td>1.60</td>
<td>3</td>
</tr>
<tr>
<td>.10</td>
<td>4</td>
<td>1.60</td>
<td>3</td>
</tr>
<tr>
<td>.06</td>
<td>5</td>
<td>1.92</td>
<td>4</td>
</tr>
<tr>
<td>.06</td>
<td>5</td>
<td>1.92</td>
<td>5</td>
</tr>
<tr>
<td>.04</td>
<td>5</td>
<td>1.28</td>
<td>5</td>
</tr>
<tr>
<td>.04</td>
<td>5</td>
<td>1.28</td>
<td>5</td>
</tr>
<tr>
<td>.04</td>
<td>5</td>
<td>1.28</td>
<td>5</td>
</tr>
<tr>
<td>.03</td>
<td>6</td>
<td>1.92</td>
<td>6</td>
</tr>
<tr>
<td>.01</td>
<td>7</td>
<td>1.28</td>
<td>6</td>
</tr>
</tbody>
</table>

In Table 1, \( \sum_{i=1}^{K} p_i = 1 \). Also \( -\log_2 p_i < n_i < 1 - \log_2 p_i \) (Definition 12), \( v_i = p_i 2^{n_i} \) (Definition 16), and \( \tilde{r}_i \) is the length of the Huffman code words.

\( S_E = 83/128 \) and so \( R_E = 2^{-7} + 2^{-5} + 2^{-4} + 2^{-2} \).

In Table 2, only the valuations \( v_i \) and the code word lengths \( n_i \) of the outer codes \( E \) are necessary; the probabilities \( p_i \) are illustrative. Common valuations are found
by halving or averaging valuations and the final Huffman encoding arranges the probabilities in descending order with the resultant code word lengths in ascending order, as in \( \{(p_i, n_i)\}_{i=1}^{K} \) of Table 1.

Table 2

<table>
<thead>
<tr>
<th></th>
<th>( E_7 )</th>
<th>( E_6 )</th>
<th>( E_5 )</th>
<th>( E_4 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( v_i )</td>
<td>( p_i )</td>
<td>( n_i )</td>
<td>( v_i )</td>
<td>( p_i )</td>
</tr>
<tr>
<td>1.92</td>
<td>.06</td>
<td>5</td>
<td>1.92</td>
<td>.06</td>
</tr>
<tr>
<td>1.92</td>
<td>.06</td>
<td>5</td>
<td>1.92</td>
<td>.06</td>
</tr>
<tr>
<td>1.92</td>
<td>.03</td>
<td>6</td>
<td>1.92</td>
<td>.03</td>
</tr>
<tr>
<td>1.60</td>
<td>.20</td>
<td>3</td>
<td>1.60</td>
<td>.20</td>
</tr>
<tr>
<td>1.60</td>
<td>.10</td>
<td>4</td>
<td>1.60</td>
<td>.10</td>
</tr>
<tr>
<td>1.60</td>
<td>.10</td>
<td>4</td>
<td>1.60</td>
<td>.10</td>
</tr>
<tr>
<td>1.60</td>
<td>.10</td>
<td>4</td>
<td>1.60</td>
<td>.10</td>
</tr>
<tr>
<td>1.44</td>
<td>.18</td>
<td>3</td>
<td>1.44</td>
<td>.18</td>
</tr>
<tr>
<td>1.28</td>
<td>.04</td>
<td>5</td>
<td>1.28</td>
<td>.04</td>
</tr>
<tr>
<td>1.28</td>
<td>.04</td>
<td>5</td>
<td>1.28</td>
<td>.04</td>
</tr>
<tr>
<td>1.28</td>
<td>.04</td>
<td>5</td>
<td>1.28</td>
<td>.04</td>
</tr>
<tr>
<td>1.28</td>
<td>.01</td>
<td>(7)</td>
<td>.64</td>
<td>.01</td>
</tr>
</tbody>
</table>

\( E_7 \): (1.28, .01, 7) is the only case to be considered. As \( 2^{-7} \) is a term of \( R_E \), 7 is reduced to 6, giving the new valuation \( .64 = 1.28 / 2 \).

\( E_6 \): (1.92, .03, 6) and (1.64, .01, 6) are combined, as \( 2^{-6} \) is not a term of \( R_E \). They appear with common valuation \( 1.28 = (1.92 + .64) / 2 \) as a 5-block in \( E_5 \) since \( 2^{-6} + 2^{-6} = 2^{-5} \).

Multiple element \( \mu \)-blocks are bracketed in \( E_5 \) and \( E_4 \).
$E_5$: (1.92, .06, 5) is reduced, for it is the first
5-block on the valuation list and $2^{-5}$ is a term of $R_E$.
The remaining 5-blocks $N_{5,\lambda}, \lambda = 2, 3, \ldots, 7$ are
combined in order of their appearance on the list, and their
common valuations are averaged.

$E_4$: The first four code word lengths are reduced by 1
since $\sum_{i=1}^{4} n_i = 2^{-4} + 2^{-2}$, and these are the terms still
to be accounted for in $R_E$. Hence, the final code word
lengths have been found ($S_c = 1$ and $L_c = 3.42$).

EXAMPLE 2: In this and subsequent examples, the probabilities
are not used after the original valuations have been computed.
However, since the $\{p_i\}_{i=1}^{K}$ are in descending order, the
$\{\tilde{n}_i\}_{i=1}^{K}$ when found are written in ascending order to give
$C = \{(p_i, \tilde{n}_i)\}_{i=1}^{K}$.

The example is tabled on the following page. As $D = 6$,
and $K = 29 \equiv 4 \pmod 5$, the last four probabilities,
namely, .004, .002, .002, and .002, are summed to $p_\alpha = .01$
[denoted by parentheses]. When the Huffman code word length
$\tilde{n}_\alpha$ for this substitute probability is found, set $\tilde{n}_i = \tilde{n}_\alpha + 1$,
$i = 26, 27, 28, 29$. The substitute probability $p_\alpha$ and its
code word length $\tilde{n}_\alpha$ is then removed from the final encoding.

The $n_i$ and $v_i$ are defined by the Shannon code:
$-\log_6 p_i \leq n_i < 1 - \log_6 p_i$ and $v_i = p_i n_i$, $i = 1, 2, \ldots, 25$
and $i = \alpha$. $S_E = 101 / 216$ or $R_E = 5(5)6^{-3} + 3(5)6^{-2}$.

Note that $(D - 1)D^{-1} = (5)6^{-1} > R_E$ and hence $n_1 = 1$
cannot be reduced. Therefore to assist calculations, $v_1$ is
given the value 0.0 so that it will appear at the bottom.
\[ D = 0 \text{ and } K = 29 \equiv 4 \pmod{5} \]

<table>
<thead>
<tr>
<th>No.</th>
<th>( p_i )</th>
<th>( n_i )</th>
<th>( v_i )</th>
<th>( \tilde{n}_i )</th>
<th>( v_{(i)} )</th>
<th>( n_{(i)} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>.20</td>
<td>1</td>
<td>1.20 = 0.6</td>
<td>1</td>
<td>5.40</td>
<td>(2) 1</td>
</tr>
<tr>
<td>2</td>
<td>.18</td>
<td>2</td>
<td>5.40</td>
<td>1</td>
<td>4.68</td>
<td>(2) 1</td>
</tr>
<tr>
<td>3</td>
<td>.13</td>
<td>2</td>
<td>4.68</td>
<td>1</td>
<td>4.68</td>
<td>(2) 1</td>
</tr>
<tr>
<td>4</td>
<td>.13</td>
<td>2</td>
<td>4.68</td>
<td>1</td>
<td>4.32</td>
<td>(3) 2</td>
</tr>
<tr>
<td>5</td>
<td>.05</td>
<td>2</td>
<td>1.80</td>
<td>2</td>
<td>4.32</td>
<td>(3) 2</td>
</tr>
<tr>
<td>6</td>
<td>.04</td>
<td>2</td>
<td>1.44</td>
<td>2</td>
<td>4.32</td>
<td>(3) 2</td>
</tr>
<tr>
<td>7</td>
<td>.04</td>
<td>2</td>
<td>1.44</td>
<td>2</td>
<td>4.32</td>
<td>(3) 2</td>
</tr>
<tr>
<td>8</td>
<td>.04</td>
<td>2</td>
<td>1.44</td>
<td>2</td>
<td>4.32</td>
<td>(3) 2</td>
</tr>
<tr>
<td>9</td>
<td>.03</td>
<td>2</td>
<td>1.08</td>
<td>2</td>
<td>2.16</td>
<td>3</td>
</tr>
<tr>
<td>10</td>
<td>.02</td>
<td>3</td>
<td>4.32</td>
<td>2</td>
<td>2.16</td>
<td>3</td>
</tr>
<tr>
<td>11</td>
<td>.02</td>
<td>3</td>
<td>4.32</td>
<td>2</td>
<td>2.16</td>
<td>3</td>
</tr>
<tr>
<td>12</td>
<td>.02</td>
<td>3</td>
<td>4.32</td>
<td>2</td>
<td>2.16</td>
<td>3</td>
</tr>
<tr>
<td>13</td>
<td>.02</td>
<td>3</td>
<td>4.32</td>
<td>2</td>
<td>2.16</td>
<td>3</td>
</tr>
<tr>
<td>14</td>
<td>.02</td>
<td>3</td>
<td>4.32</td>
<td>2</td>
<td>2.16</td>
<td>3</td>
</tr>
<tr>
<td>15</td>
<td>.01</td>
<td>3</td>
<td>2.16</td>
<td>3</td>
<td>1.80</td>
<td>2</td>
</tr>
<tr>
<td>16</td>
<td>.01</td>
<td>3</td>
<td>2.16</td>
<td>3</td>
<td>1.44</td>
<td>2</td>
</tr>
<tr>
<td>17</td>
<td>.01</td>
<td>3</td>
<td>2.16</td>
<td>3</td>
<td>1.44</td>
<td>2</td>
</tr>
<tr>
<td>18</td>
<td>.01</td>
<td>3</td>
<td>2.16</td>
<td>3</td>
<td>1.44</td>
<td>2</td>
</tr>
<tr>
<td>19</td>
<td>.01</td>
<td>3</td>
<td>2.16</td>
<td>3</td>
<td>1.08</td>
<td>2</td>
</tr>
<tr>
<td>(a)</td>
<td>(.01)</td>
<td>3</td>
<td>2.16</td>
<td>(3)</td>
<td>1.08</td>
<td>3</td>
</tr>
</tbody>
</table>

\[ S_C = \frac{1294}{1296} \text{ and } L_C = 1.49 \]
of the valuation list.

In $E_3$, step (iii) of the algorithm applies immediately, reducing the first eight $n_i$ by 1 as indicated by the parentheses. As illustrated here, step (iii) may apply at an early stage of the algorithm; in particular, this often happens when there are a number of equal probabilities.

Finally, $C = \{(p_i, \tilde{n}_i)\}_{i=1}^{29}$ is found by ordering $\tilde{n}_1 \leq \tilde{n}_2 \leq \ldots \leq \tilde{n}_\alpha \leq \ldots \leq \tilde{n}_{25}$ and setting $\tilde{n}_i = 3 + 1$, $i = 26, 27, 28, 29$.

Because $K \not\equiv 1 \pmod{D-1}$, $S_C < 1$. END OF EXAMPLE.

Three results are given which can assist in the application of the algorithm. It is assumed that $K \equiv 1 \pmod{D-1}$.

RESULT 1: The following are equivalent:

(I) $-\log_D p_i \leq n_i < 1 - \log_D p_i$

(II) $D^{-n_i} > p_i > D^{-n_i}$

(III) $1 < v_i < D$ where $v_i = p_i D^{n_i}$

By (II), for a given set of probabilities $\{p_i\}_{i=1}^K$, a sequence of inequalities is formed with a parallel sequence for code word lengths $\{n_i\}_{i=1}^K$:

$p_a > D^{-1} > p_b > D^{-2} > p_c > D^{-3} > \ldots$

$n_a = 1; \; n_b = 2; \; n_c = 3; \; \ldots$
By (III), a system for finding the \( n_i \) and \( v_i \) for \( \{p_i\}_{i=1}^K \) is constructed:

Let \( r = 1 \), and \( v_i^r = p_i D^F \). If \( v_i^r < 1 \), increase \( r \) until \( 1 \leq v_i^r < D \); then \( n_i = r \) and \( v_i = v_i^r \).

If \( p_1 > p_2 > \ldots > p_K \) for a Shannon code \( \{(p_i, n_i)\}_{i=1}^K \), then \( n_1 \leq n_2 \leq \ldots \leq n_K \) and either of the above suggested methods may be used successively for \( p_i \), \( i = 1, 2, \ldots, K \), to find \( n_i \).

RESULT 2: Let \( E = \{(p_i, n_i)\}_{i=1}^K \) be a Shannon code with

\[
R_E = \sum_{\mu=1}^{n_K} a_\mu (D - 1)D^{-\mu}, \quad 0 \leq a_\mu < D.
\]

Let \( Q_\mu \) be the number of \( n_i = \mu \), \( \mu = n_K, n_K - 1, \ldots, 2, 1 \). Then for all \( \mu \), \( n_K > \mu > 1 \), there exists an integer \( b_\mu \) such that

\[
Q_{n_K} = b_{n_K} D + a_{n_K}
\]

and \( Q_\mu + b_{\mu+1} D + a_{\mu+1} = b_\mu D + a_\mu \); \( n_K > \mu > 1 \).

In fact, since \( 0 \leq a_\mu < D \), then \( b_{n_K} = \lfloor Q_{n_K} / D \rfloor \) and

\[
b_\mu = \left(\lfloor Q_\mu + b_{\mu+1} D + a_{\mu+1} \rfloor / D\right), \quad n_K > \mu > 1.
\]

Proof: As in Lemma 12, there are three cases to be considered:

(a) \( a_{\mu+1} \) is the number of \( \mu \)-blocks formed by reducing \( (\mu+1) \)-blocks, \( n_K > \mu > 1 \).

(b) \( b_{\mu+1} \) is the number of \( \mu \)-blocks formed by combining \( (\mu+1) \)-blocks \( D \) at a time, \( n_K > \mu > 1 \).

(c) \( Q_\mu \) is the number of \( \mu \)-blocks \( \{n_i = \mu\} \) not reduced from \( E \), \( n_K > \mu > 1 \).
Therefore, let \( m_\mu \) be the total number of \( \mu \)-blocks. Then \( m_{\tilde{n}} = Q_{\tilde{n}}, \) and \( m_\mu = Q_\mu + b_\mu + a_\mu + 1, \) \( 1 \leq \mu \leq n_K. \)

By Lemma 11, \( D | (m_\mu - a_\mu), \) \( 1 \leq \mu \leq n_K. \) QED

By means of this result, an algorithm can be constructed to find the \( a_\mu, n_K \geq \mu \geq 1, \) of \( R_E. \)

RESULT 3: Let \( M = \{(p_i, \tilde{n}_i)\}_{i=1}^K \) be a Huffman code with \( K \equiv 1 \pmod{D-1}. \) Then

\[
\tilde{n}_{K-D+1} = \tilde{n}_{K-D+2} = \ldots = \tilde{n}_K \leq \tilde{n}_{K-D} + 1.
\]

Proof: This is immediate from Huffman's procedure \([15]\). QED

This result permits a defining of a new original outer code. Let \( E = \{(p_i, n_i)\}_{i=1}^K \) be a Shannon code, and let \( n = n_{K-D} + 1. \) Define a code \( E' = \{(p_i, n'_i)\}_{i=1}^K \) by:

\[
n'_i = n_i, \ i = 1, 2, \ldots, K-D
\]

\[
n'_i = \min\{n_{K-D+1}, n\}, \ i = K-D+1, K-D+2, \ldots, K.
\]

By Theorem 3 and Result 3, \( \tilde{n}_i \leq n'_i, \ i = 1, 2, \ldots, K; \) so \( E' \) can serve as the original outer code.

EXAMPLE 3: (The example is tabled on the following page.)

The \( \{p_i\}_{i=1}^K \) are listed in descending order. By means of Result 1, a sequence is set up:

\[
.25 > p_a \geq .125 > p_b \geq .0625 > p_c \geq .03125 > \ldots
\]

\[
n_a = 3; \ n_b = 4; \ n_c = 5; \ \ldots
\]

Result 3 permits \((.003, 8)\) and \((.0015, 8)\) to be used in the original outer code. Then \( v_i = p_i^2, i = K-1, K. \)
<table>
<thead>
<tr>
<th>Shannon Code</th>
<th>Huffman Code</th>
<th>Valuation List</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p_i$</td>
<td>$n_i$</td>
<td>$v_i$</td>
</tr>
<tr>
<td>.19</td>
<td>3</td>
<td>1.520</td>
</tr>
<tr>
<td>.16</td>
<td>3</td>
<td>1.280</td>
</tr>
<tr>
<td>.12</td>
<td>4</td>
<td>1.920</td>
</tr>
<tr>
<td>.10</td>
<td>4</td>
<td>1.600</td>
</tr>
<tr>
<td>.06</td>
<td>5</td>
<td>1.920</td>
</tr>
<tr>
<td>.045</td>
<td>5</td>
<td>1.440</td>
</tr>
<tr>
<td>.04</td>
<td>5</td>
<td>1.280</td>
</tr>
<tr>
<td>.04</td>
<td>5</td>
<td>1.280</td>
</tr>
<tr>
<td>.04</td>
<td>5</td>
<td>1.280</td>
</tr>
<tr>
<td>.035</td>
<td>5</td>
<td>1.120</td>
</tr>
<tr>
<td>.03</td>
<td>6</td>
<td>1.920</td>
</tr>
<tr>
<td>.03</td>
<td>6</td>
<td>1.920</td>
</tr>
<tr>
<td>.02</td>
<td>6</td>
<td>1.280</td>
</tr>
<tr>
<td>.02</td>
<td>6</td>
<td>1.280</td>
</tr>
<tr>
<td>.014</td>
<td>7</td>
<td>1.792</td>
</tr>
<tr>
<td>.0115</td>
<td>7</td>
<td>1.472</td>
</tr>
<tr>
<td>.01</td>
<td>7</td>
<td>1.280</td>
</tr>
<tr>
<td>.01</td>
<td>7</td>
<td>1.280</td>
</tr>
<tr>
<td>.01</td>
<td>7</td>
<td>1.280</td>
</tr>
<tr>
<td>.003</td>
<td>9 ≈ 8</td>
<td>.768</td>
</tr>
<tr>
<td>.0015</td>
<td>10 ≈ 8</td>
<td>.384</td>
</tr>
</tbody>
</table>

\[
\mu \quad Q_\mu \quad (b_{\mu+1} + a_{\mu+1}) \quad b_\mu \cdot D \quad a_\mu
\]

| 8 | 2 | = | 1.2 + 0 |
| 7 | 6 + 1 | = | 3.2 + 1 |
| 6 | 4 + 4 | = | 4.2 + 0 |
| 5 | 6 + 4 | = | 5.2 + 0 |
| 4 | 2 + 5 | = | 3.2 + 1 |
| 3 | 2 + 4 | = | 3.2 + 0 |
| 2 | 0 + 3 | = | 1.2 + 1 |
| 1 | 0 + 2 | = | 1.2 + 0 |

\[
P_E = 2^{-7} + 2^{-4} + 2^{-2}
\]
By Result 2, \( a_\mu \), \( n_K \geq \mu \geq 1 \), is found by means of an algorithm. Here, \( Q_\mu \) is the number of singleton sets \( \{n_1 = \mu\} \) in the original outer code, \( b_\mu = [Q_\mu + (b_{\mu+1} + a_{\mu+1})] \), and \( a_\mu \equiv (Q_\mu + b_{\mu+1} + a_{\mu+1})(\text{mod } 2) \), \( 0 \leq a_\mu \leq 2 \). Then

\[
R_E = \sum_{\mu=1}^{n_K} a_\mu (D - 1)D^{-\mu}.
\]

Then the valuation list \( \{(v_i, n_i)\}_{i=1}^{K} \) is ordered such that \( v_1 \geq v_2 \geq \ldots \geq v_K \). Step (iii) of the algorithm applies immediately and since \( R_E = \sum_{i=1}^{7} 2^{-n_i} \) of the valuation list, the first seven \( n_i \) are reduced by 1. The Huffman code is formed as before, ordering the \( \{\tilde{n}_i\}_{i=1}^{K} \) with \( \tilde{n}_1 \leq \tilde{n}_2 \leq \ldots \leq \tilde{n}_K \).

**EXAMPLE 4:** \( D = 5 \), \( K = 17 \equiv 1 \pmod{4} \)

<table>
<thead>
<tr>
<th>Shannon Code</th>
<th>Huffman Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>( p_i )</td>
<td>( \tilde{n}_i )</td>
</tr>
<tr>
<td>.23</td>
<td>1</td>
</tr>
<tr>
<td>.18</td>
<td>2</td>
</tr>
<tr>
<td>.09</td>
<td>2</td>
</tr>
<tr>
<td>.08</td>
<td>2</td>
</tr>
<tr>
<td>.07</td>
<td>2</td>
</tr>
<tr>
<td>.0625</td>
<td>2</td>
</tr>
<tr>
<td>.058</td>
<td>2</td>
</tr>
<tr>
<td>.03</td>
<td>3</td>
</tr>
<tr>
<td>.03</td>
<td>2</td>
</tr>
<tr>
<td>.0275</td>
<td>.1365, 2</td>
</tr>
<tr>
<td>.0265</td>
<td>2</td>
</tr>
<tr>
<td>.0225</td>
<td>2</td>
</tr>
<tr>
<td>.0216</td>
<td>2</td>
</tr>
<tr>
<td>.021</td>
<td>2</td>
</tr>
<tr>
<td>.0186</td>
<td>.0930, 2</td>
</tr>
<tr>
<td>.018</td>
<td>2</td>
</tr>
<tr>
<td>.018</td>
<td>2</td>
</tr>
<tr>
<td>.0138</td>
<td>2</td>
</tr>
</tbody>
</table>
\[
\begin{align*}
\mu & \quad Q_{\mu} & \quad (b_{\mu+1} + a_{\mu+1}) & \quad b_{\mu} \cdot D & \quad a_{\mu} \\
3 & \quad 15 & \quad = & \quad 3 \cdot 5 & \quad + & \quad 0 \\
2 & \quad 5 & \quad + & \quad 3 & \quad = & \quad 1 \cdot 5 & \quad + & \quad 3 \\
1 & \quad 1 & \quad + & \quad 4 & \quad = & \quad 1 \cdot 5 & \quad + & \quad 0 \\
\end{align*}
\]

\[R_{E} = 3(4)5^{-2}\]

In this example, it was not necessary to take valuations.

As \(R_{E}\) is a simple expression, and the 2-blocks can easily be found, inspection of the Shannon code immediately gives the Huffman code. The three 2-blocks with highest total probability are reduced.

END OF EXAMPLE.

Another form of the algorithm does not calculate the common valuations of all \(\mu\)-blocks, but restricts itself to \(\mu\)-blocks of high valuation, \(\mu = n_K, n_K - 1, \ldots, 2, 1\).

It investigates the original outer code \(E_{n_K}\), and selects sufficient \(n_i \geq \mu\) to form the \(a_{\mu}\) \(\mu\)-blocks of highest valuation. By Lemma 12, these are also the \(a_{\mu}\) \(\mu\)-blocks of highest total probability. All the \(n_{ij}\) in the selected \(\mu\)-blocks are replaced by \(n_{ij}' = n_{ij} - 1\), written one space to the right of \(n_{ij}\). The moving of \(n_{ij}'\) one column to the right is equivalent to setting its valuations \(v_{ij}' = v_{ij} / D\).

The \(n_i\) which have not been reduced have in the first column \(v_1 \geq 1\), in the second \(1 > v_1 > D^{-1}\), in the third \(D^{-1} > v_1 > D^{-2}\), etc. Thus the \(n_i\) remain in order of valuation if one proceeds down the first column, then the second, third, etc., ignoring the reduced \(n_i\).
Any \( n_1 \) reduced by means of Result 3 is found in the original outer code and hence appears in the first column. Since the valuations of such \( n_1 \) are less than 1, these \( n_1 \) properly belong in the second, third, or following columns. But \( v_1 < 1 \) places them at the bottom of the first column. Therefore they are reduced only after the \( n_{K-D} \) of the Shannon code is reduced. By Result 3, this is a legitimate procedure. Thus these \( n_1 \) may remain in the first column until further reduced.

EXAMPLE 5:

**Huffman Encoding for the Dewey Distribution of the English Alphabet**

(The example is tabled on the two following pages.)

**Phase 1:** Since \( 2^{-11} \) is a term of \( R_E \), an 11-block must be reduced. (1.6384, .0008, 11) is the first 11-block on the list and so has the highest valuation of all 11-blocks. Writing (11) 10 represents the division of \( v_1 = 1.6384 \) by 2.

To subtract \( 2^{-10} \) from \( R_E \), a 10-block is reduced. The first listed \( n_1 \geq 10 \) are marked with a * . Only the first three \( n_1 \geq 10 \) are so marked, as (1.6384, .0008, 11) and (1.0240, .0005, 11) form a 10-block which includes the \( n_1 \geq 10 \) of highest valuation. The unmarked 10-blocks then have valuations no greater than this one. Also (1.3312, .0013, 10) is a 10-block in itself and needs only to be compared to 10-blocks composed of one or more \( n_1 \geq 10 \) of higher valuation.
\[ D = 2 \]

<table>
<thead>
<tr>
<th>Character</th>
<th>( p_i )</th>
<th>( n_i )</th>
<th>( v_i )</th>
<th>( \tilde{n}_i )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Space</td>
<td>0.1859</td>
<td>3</td>
<td>1.4872</td>
<td>3</td>
</tr>
<tr>
<td>E</td>
<td>0.1031</td>
<td>4</td>
<td>1.6496</td>
<td>3</td>
</tr>
<tr>
<td>T</td>
<td>0.0796</td>
<td>4</td>
<td>1.2736</td>
<td>4</td>
</tr>
<tr>
<td>A</td>
<td>0.0642</td>
<td>4</td>
<td>1.0272</td>
<td>4</td>
</tr>
<tr>
<td>O</td>
<td>0.0632</td>
<td>4</td>
<td>1.0112</td>
<td>4</td>
</tr>
<tr>
<td>I</td>
<td>0.0575</td>
<td>5</td>
<td>1.8400</td>
<td>4</td>
</tr>
<tr>
<td>N</td>
<td>0.0574</td>
<td>5</td>
<td>1.8368</td>
<td>4</td>
</tr>
<tr>
<td>S</td>
<td>0.0514</td>
<td>5</td>
<td>1.6448</td>
<td>4</td>
</tr>
<tr>
<td>R</td>
<td>0.0484</td>
<td>5</td>
<td>1.5488</td>
<td>4</td>
</tr>
<tr>
<td>H</td>
<td>0.0467</td>
<td>5</td>
<td>1.4944</td>
<td>4</td>
</tr>
<tr>
<td>L</td>
<td>0.0321</td>
<td>5</td>
<td>1.0272</td>
<td>5</td>
</tr>
<tr>
<td>D</td>
<td>0.0317</td>
<td>5</td>
<td>1.0144</td>
<td>5</td>
</tr>
<tr>
<td>U</td>
<td>0.0228</td>
<td>6</td>
<td>1.4592</td>
<td>5</td>
</tr>
<tr>
<td>C</td>
<td>0.0218</td>
<td>6</td>
<td>1.3952</td>
<td>5</td>
</tr>
<tr>
<td>F</td>
<td>0.0208</td>
<td>6</td>
<td>1.3312</td>
<td>6</td>
</tr>
<tr>
<td>M</td>
<td>0.0198</td>
<td>6</td>
<td>1.2672</td>
<td>6</td>
</tr>
<tr>
<td>W</td>
<td>0.0175</td>
<td>6</td>
<td>1.1200</td>
<td>6</td>
</tr>
<tr>
<td>Y</td>
<td>0.0164</td>
<td>6</td>
<td>1.0496</td>
<td>6</td>
</tr>
<tr>
<td>G</td>
<td>0.0152</td>
<td>7</td>
<td>1.9456</td>
<td>6</td>
</tr>
<tr>
<td>P</td>
<td>0.0152</td>
<td>7</td>
<td>1.9456</td>
<td>6</td>
</tr>
<tr>
<td>B</td>
<td>0.0127</td>
<td>7</td>
<td>1.6256</td>
<td>6</td>
</tr>
<tr>
<td>V</td>
<td>0.0083</td>
<td>7</td>
<td>1.0624</td>
<td>7</td>
</tr>
<tr>
<td>K</td>
<td>0.0049</td>
<td>8</td>
<td>1.2544</td>
<td>8</td>
</tr>
<tr>
<td>X</td>
<td>0.0013</td>
<td>10</td>
<td>1.3312</td>
<td>10</td>
</tr>
<tr>
<td>J</td>
<td>0.0008</td>
<td>11</td>
<td>1.6384</td>
<td>10</td>
</tr>
<tr>
<td>Q</td>
<td>0.0008</td>
<td>11</td>
<td>1.6384</td>
<td>10</td>
</tr>
<tr>
<td>Z</td>
<td>0.0005</td>
<td>11</td>
<td>1.0240</td>
<td>10</td>
</tr>
</tbody>
</table>

\[
R_E = 2^{-11} + 2^{-10} + 2^{-7} + 2^{-6} + 2^{-2}
\]
Valuation List

<table>
<thead>
<tr>
<th>$v_i$</th>
<th>$p_i$</th>
<th>$n_i$</th>
<th>Phase 1</th>
<th>Phase 2</th>
<th>Phase 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.9456</td>
<td>.0152</td>
<td>7</td>
<td>(7) 6</td>
<td>(7) 6</td>
<td></td>
</tr>
<tr>
<td>1.9456</td>
<td>.0152</td>
<td>7</td>
<td>(7) 6</td>
<td>(7) 6</td>
<td></td>
</tr>
<tr>
<td>1.8400</td>
<td>.0574</td>
<td>5</td>
<td>5 *</td>
<td>(5) 4</td>
<td></td>
</tr>
<tr>
<td>1.8368</td>
<td>.0574</td>
<td>5</td>
<td>5 *</td>
<td>(5) 4</td>
<td></td>
</tr>
<tr>
<td>1.6496</td>
<td>.1031</td>
<td>4</td>
<td>4 *</td>
<td>(4) 3</td>
<td></td>
</tr>
<tr>
<td>1.6448</td>
<td>.0514</td>
<td>5</td>
<td>5 *</td>
<td>(5) 4</td>
<td></td>
</tr>
<tr>
<td>1.6384</td>
<td>.0008</td>
<td>(11) 10</td>
<td>(11) 10</td>
<td>(11) 10</td>
<td></td>
</tr>
<tr>
<td>1.6384</td>
<td>.0008</td>
<td>11 *</td>
<td>(11) 10</td>
<td>(11) 10</td>
<td></td>
</tr>
<tr>
<td>1.6256</td>
<td>.0127</td>
<td>7</td>
<td>(7) 6</td>
<td>(7) 6</td>
<td></td>
</tr>
<tr>
<td>1.5488</td>
<td>.0484</td>
<td>5</td>
<td>5 *</td>
<td>(5) 4</td>
<td></td>
</tr>
<tr>
<td>1.4944</td>
<td>.0467</td>
<td>5</td>
<td>5 *</td>
<td>(5) 4</td>
<td></td>
</tr>
<tr>
<td>1.4872</td>
<td>.1859</td>
<td>3</td>
<td>3 *</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>1.4592</td>
<td>.0228</td>
<td>6</td>
<td>6 *</td>
<td>(6) 5</td>
<td></td>
</tr>
<tr>
<td>1.3952</td>
<td>.0218</td>
<td>6</td>
<td>6 *</td>
<td>(6) 5</td>
<td></td>
</tr>
<tr>
<td>1.3312</td>
<td>.0208</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>1.3312</td>
<td>.0013</td>
<td>10 *</td>
<td>10</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>1.2736</td>
<td>.0796</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>1.2672</td>
<td>.0198</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>1.2544</td>
<td>.0049</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>1.1200</td>
<td>.0164</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>1.0624</td>
<td>.0083</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>1.0496</td>
<td>.0164</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>1.0272</td>
<td>.0642</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>1.0272</td>
<td>.0321</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>1.0240</td>
<td>.0005</td>
<td>(11) 10</td>
<td>(11) 10</td>
<td>(11) 10</td>
<td></td>
</tr>
<tr>
<td>1.0144</td>
<td>.0317</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>1.0112</td>
<td>.0632</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td></td>
</tr>
</tbody>
</table>
From these marked \( n_1 \geq 10 \), two 10-blocks can be formed:

\[
1.3312 \quad .0008 \quad 11 \quad 1.3312 \quad .0013 \quad 10 \\
1.3312 \quad .0005 \quad 11
\]

Since the common valuations and total probabilities are equal, the choice between the two 10-blocks is arbitrary. It will prove advantageous to reduce the one with the most \( n_1 \geq 10 \).

Phase 2: Here \( 2^{-7} \) can be immediately subtracted from \( R_E \) by reducing \((1.9456, .0152, 7)\), the first 7-block on the list.

Similarly \( 2^{-6} \) is subtracted as

\[
(1.9456, .0152, 7) \\
(1.6256, .0127, 7)
\]

is the first 6-block on the list.

For \( 2^{-2} \), a choice must be made from the \( n_1 \geq 2 \) marked by a * . There are two 2-blocks to be considered since the first 3-block on the list is obvious.

\[
.0575 \quad 5 \\
.0574 \quad 5 \\
.1031 \quad 4 \\
.0514 \quad 5 \\
.0484 \quad 5 \\
.0467 \quad 5 \\
.0228 \quad 6 \\
.0218 \quad 6
\]

The left is equivalent to \((.4091, 2)\) and the right to \((.4039, 2)\). Hence each \( n_1 \geq 2 \) in the right-hand 2-block is reduced by 1. As \( R_E = 0 \), the Huffman encoding is given in Phase 3.

END OF EXAMPLE.
Schwartz [29] has pointed out the desirability of having the largest Huffman code word length \( \tilde{r}_K \) and the sum of the code word lengths \( \sum_{i=1}^{K} \tilde{r}_i \) as short as possible.

RESULT 4: Let \( \{(v_i, p_i, n_i)\}_{i=1}^{K} \) be an outer code. To minimize \( \tilde{r}_K \) and \( \sum_{i=1}^{K} \tilde{r}_i \) of the Huffman code \( C = \{(p_i, \tilde{r}_i)\}_{i=1}^{K} \), list all \( (v_i, p_i, n_i) \) of equal valuation in descending order of \( n_i \). Then apply the algorithm. Proof: Though the Shannon code, and hence the original outer code, is unique for a fixed set of probabilities \( \{p_i\}_{i=1}^{K} \), the Huffman encoding need not be. This occurs when a choice must be made to reduce \( n_i \) of equal valuations. Since \( n_K > n_{K-1} > \cdots > n_1 \), by listing the \( n_i \) of equal valuation in descending order, \( \tilde{r}_K \) will be reduced whenever its place on the valuation list requires it. That is, \( \tilde{r}_K \) will be as short as possible. Also, reducing the largest \( n_i \) gives the smallest increase \( D^{-n_i} \) in \( S_E \); so the total number of reductions is maximized. Therefore \( \sum_{i=1}^{K} \tilde{r}_i \) is a minimum for a Huffman encoding \( C \). QED

Let \( \{(p_i, n_i)\}_{i=1}^{K} \) be a Shannon code with \( p_1 \geq p_2 \geq \cdots \geq p_K \). In practice, Result 4 is implemented by listing the valuations of \( n_K \) such that \( v_1 \geq v_2 \geq \cdots \geq v_K \) by some method that examines in turn \( (v_i, p_i, n_i) \), \( i = K, K-1, \ldots, 2, 1 \). Since \( n_K > n_{K-1} > \cdots > n_2 > n_1 \), the \( (v_i, p_i, n_i) \) of equal valuation are listed in order from the largest \( n_i \) to the smallest.
By this ordering of $E_{n_K}$, the $\mu$-blocks with the most members also have the largest values of $n_i$; hence, either criterion can be used in listing the $\mu$-blocks with equal common valuations.

**EXAMPLE 6:** (The $\{p_i\}_{i=1}^K$ of Example 1 are encoded using Results 1, 2, 3, and 4.)

$D = 2$. 

<table>
<thead>
<tr>
<th>Shannon Code</th>
<th>Huffman Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p_i$</td>
<td>$n_i$</td>
</tr>
<tr>
<td>.20</td>
<td>3</td>
</tr>
<tr>
<td>.18</td>
<td>3</td>
</tr>
<tr>
<td>.10</td>
<td>4</td>
</tr>
<tr>
<td>.10</td>
<td>4</td>
</tr>
<tr>
<td>.10</td>
<td>4</td>
</tr>
<tr>
<td>.06</td>
<td>5</td>
</tr>
<tr>
<td>.06</td>
<td>5</td>
</tr>
<tr>
<td>.04</td>
<td>5</td>
</tr>
<tr>
<td>.04</td>
<td>5</td>
</tr>
<tr>
<td>.04</td>
<td>5</td>
</tr>
<tr>
<td>.03</td>
<td>6</td>
</tr>
<tr>
<td>.03</td>
<td>7</td>
</tr>
</tbody>
</table>

$$
\begin{align*}
\mu & \quad Q_\mu \quad (b_{\mu+1} + a_{\mu+1}) \quad b_\mu \cdot D \quad a_\mu \\
6 & \quad 2 \quad = \quad 1 \cdot 2 + 0 \\
5 & \quad 6 \quad + \quad 1 \quad = \quad 3 \cdot 2 + 1 \\
4 & \quad 3 \quad + \quad 4 \quad = \quad 3 \cdot 2 + 1 \\
3 & \quad 2 \quad + \quad 4 \quad = \quad 3 \cdot 2 + 0 \\
2 & \quad 0 \quad + \quad 3 \quad = \quad 1 \cdot 2 + 1 \\
1 & \quad 0 \quad + \quad 2 \quad = \quad 1 \cdot 2 + 0 \\
\end{align*}
$$

$R_E = 2^{-5} + 2^{-4} + 2^{-2}$
### Valuation List

<table>
<thead>
<tr>
<th>$v_i$</th>
<th>$p_i$</th>
<th>$n_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Phase 1</td>
</tr>
<tr>
<td>1.92</td>
<td>.03</td>
<td>6 *</td>
</tr>
<tr>
<td>1.92</td>
<td>.06</td>
<td>5 *</td>
</tr>
<tr>
<td>1.92</td>
<td>.06</td>
<td>5</td>
</tr>
<tr>
<td>1.60</td>
<td>.10</td>
<td>4</td>
</tr>
<tr>
<td>1.60</td>
<td>.10</td>
<td>4</td>
</tr>
<tr>
<td>1.60</td>
<td>.10</td>
<td>4</td>
</tr>
<tr>
<td>1.60</td>
<td>.20</td>
<td>3</td>
</tr>
<tr>
<td>1.44</td>
<td>.18</td>
<td>3</td>
</tr>
<tr>
<td>1.28</td>
<td>.04</td>
<td>5</td>
</tr>
<tr>
<td>1.28</td>
<td>.04</td>
<td>5</td>
</tr>
<tr>
<td>1.28</td>
<td>.04</td>
<td>5</td>
</tr>
<tr>
<td>.64</td>
<td>.01</td>
<td>6 *</td>
</tr>
</tbody>
</table>

In Example 1: $\hat{n}_K = 6$; $\sum_{i=1}^{K} \hat{n}_i = 55$.

In Example 6: $\hat{n}_K = 5$; $\sum_{i=1}^{K} \hat{n}_i = 53$.

In both examples: $S_C = 1$, $L_C = 3.42$.

### CODING WITH CONSTRAINTS

Karp [16] suggests the problem of constructing instantaneous codes $A = \{(p_i, \hat{n}_i)\}_{i=1}^{K}$ subject to "constraints" and having least average length. These constraints normally are dictated by physical limitations of the encoder and the decoder. One case is an upper bound, $N \geq 1$, placed on the code word lengths; that is, $\hat{n}_i \leq N$, $i = 1, 2, \ldots, K$. 
To find $A$, a procedure analogous to the algorithm for minimal encoding is used. Assume that $p_1 \geq p_2 \geq \cdots \geq p_K$, and that $\log_D K \leq N < n_K$, for $n_K$ from the Shannon code, $E = \{(p_i, n_i)\}_{i=1}^K$. Let the original outer code be defined by $E_N = \{(p_i, N)\}_{i=1}^K$. Then $S_{E_N} \leq 1$ (or $R_{E_N} > 0$), and the code word lengths with highest valuation are reduced until $R_A = 0$ is reached.

Usually this procedure can be shortened by defining the original outer code as $E' = \{(p_i, n_i') : n_i' = \min[n_i, N]\}_{i=1}^K$. For if $R_{E'} > 0$, then $E_N$ reduces to $E'$ in a manner similar to Theorem 2. Finally, reduce the $n_i'$ in $E'$ until $R_{E'} = 0$.

**EXAMPLE 7:** $D = 3$, $K = 17 \equiv 1 \pmod{2}$, $N = 3$.

<table>
<thead>
<tr>
<th>$p_i$</th>
<th>$n_i$</th>
<th>$n_i'$</th>
<th>$\tilde{n}_i$</th>
<th>$\tilde{n}_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>.18</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>.12</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>.12</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>.08</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>.075</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>.07</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>.06</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>.05</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>.05</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>.045</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>.04</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>.03</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>.03</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>.02</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>.01</td>
<td>5</td>
<td>3</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>.01</td>
<td>5</td>
<td>3</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>.01</td>
<td>5</td>
<td>3</td>
<td>3</td>
<td>4</td>
</tr>
</tbody>
</table>

$L_{E'} = 2.425$, $L_C = 2.385$
CHAPTER IV

ENTROPY and HUFFMAN ENCODING

Shannon's "Theorem on Noiseless Coding" (Lemma 4) states that the capacity of a channel can be made arbitrarily close to, though not less than, the entropy of a source, \( Z = \{(\alpha_i, p_i)\}_{i=1}^K \). This is accomplished by encoding sequences of two or more source symbols rather than encoding each symbol emitted. Let \( C = \{(s_i, \tilde{r}_i)\}_{i=1}^K \) be a Huffman code for \( Z \), and \( C^n \) a Huffman encoding of \( Z^n \), the \( n \)-th extension of \( Z \). Abramson [1, p.87] notes that \( L_C, L_{C^2}/2, L_{C^3}/3, \ldots \) rapidly approach \( H_D(Z) \). Typically \( L_C \) or at most \( L_{C^2}/2 \) closely approximate \( H_D(Z) \). We wish to illustrate why Huffman encoding is efficient in this respect.

THEOREM 4: Let \( C = \{(p_i, \tilde{r}_i)\}_{i=1}^K \) and \( E = \{(p_i, n_i)\}_{i=1}^K \) be the Huffman and Shannon codes, respectively, for a zero-memory source \( Z = \{(\alpha_i, p_i)\}_{i=1}^K \) with entropy \( H_D(Z) \).

Let \( r_i = n_i - \tilde{r}_i \), \( i = 1, 2, \ldots, K \). Then

\[
L_C - H_D(Z) = \sum_{i=1}^K p_i \left( \log_D n_i - r_i \right).
\]

Proof: By Theorem 3, \( \tilde{r}_i \leq n_i \); hence \( r_i > 0 \), \( i = 1, 2, \ldots, K \).

Note the identity:

\[
n_i = \log_D \left[ \left( p_i^{n_i} \right) / p_i \right] = \log_D n_i - \log_D p_i, \quad i = 1, 2, \ldots, K.
\]

Then \( L_C - H_D(Z) = \sum_{i=1}^K \tilde{r}_i p_i - \left( -\sum_{i=1}^K p_i \log_D p_i \right) \)

\[
= \sum_{i=1}^K p_i (\tilde{r}_i + \log_D p_i) = \sum_{i=1}^K p_i (n_i - r_i + \log_D p_i)
\]

\[
= \sum_{i=1}^K p_i (\log_D n_i - r_i).
\]

QED
Hence, if $r_i$ is approximately equal to $\log_D v_i$, then $L_c - H_D(Z)$ is small, and the Huffman encoding $C$ is efficient.

Suppose for some $j, 1 \leq j \leq K$, that $r_j$ does not "approximate" $\log_D v_j$. Since $0 \leq \log_D v_j < 1$, assume that $r_j > 2$. Then $\tilde{n}_j = n_j - r_j$; thus $n'_j = n_j - 1$ with valuation $v'_j = p_j/D^{1-n'_j} < 1$ has been reduced. This implies that the decrease $p_j$ in $L_E$ when we reduce $n'_j$ is less than the increase $D^{1-n'_j}$ in $S_E$. Should this occur many times, the Huffman code proves to be inefficient.

Thus an efficient Huffman encoding occurs whenever $n_i - \tilde{n}_i$ approximately equals $\log_D v_i$, $i = 1, 2, \ldots, K$. Since $\log_D v_i = n_i + \log_D p_i$, then $\tilde{n}_i$ approximately equals $-\log_D p_i$. Unlike the Shannon code, though, $\tilde{n}_i$ can be greater than, equal to, or less than, $-\log_D p_i$.

One cause of inefficient Huffman encoding is illustrated in the following example. Here, for a source $Z$ with probabilities $\{p_i\}_{i=1}^6$, $p_1 = .80$, $v_1 = 1.60$, $\log_2 v_1 = .678$, but $n_1$ cannot be reduced. Consequently, $n_i$, $i = 2, 3, 4, 5, 6$ are reduced twice giving an inefficient Huffman code $C$. However, by taking the Huffman encoding, $C^2$, of $Z^2$, the second extension of $Z$, $p_1 = .64$, $v_1 = 1.28$, $\log_2 v_1 = .356$. Then $\tilde{n}_1 - n_1 = 0$ more closely approximates $\log_2 v_1$, and only three of the $n_i$, $i = 2, 3, \ldots, 36$ are reduced twice. $C^2$ can be considered efficient.
**EXAMPLE 8:**

<table>
<thead>
<tr>
<th>D = 2</th>
<th>Z</th>
<th>E</th>
<th>C</th>
<th>Z²</th>
<th>E²</th>
<th>C²</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>P₁</td>
<td>n₁</td>
<td>t₁</td>
<td>P₁</td>
<td>n₁</td>
<td>t₁</td>
</tr>
<tr>
<td>.80</td>
<td>1</td>
<td>1</td>
<td></td>
<td>.64</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>.06</td>
<td>5</td>
<td>3</td>
<td></td>
<td>.048</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>.05</td>
<td>5</td>
<td>3</td>
<td></td>
<td>.048</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>.04</td>
<td>5</td>
<td>3</td>
<td></td>
<td>.04</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>.03</td>
<td>6</td>
<td>4</td>
<td></td>
<td>.04</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>.02</td>
<td>6</td>
<td>4</td>
<td></td>
<td>.032</td>
<td>5</td>
<td>5</td>
</tr>
</tbody>
</table>

\[
L_C = 1.45
\]
\[
L_{C^2}/2 = 1.20
\]
\[
H_D(Z) = 1.17
\]
\[
\frac{H_D(Z)}{L_C} = .807
\]
\[
\frac{H_D(Z)}{L_{C^2}/2} = .975
\]
<table>
<thead>
<tr>
<th>Definition</th>
<th>No.</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>average length</td>
<td>11</td>
<td>8</td>
</tr>
<tr>
<td>block code</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>change in the average length</td>
<td>21</td>
<td>28</td>
</tr>
<tr>
<td>change in the average length($E_{\mu}$)</td>
<td>20</td>
<td>27</td>
</tr>
<tr>
<td>code</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>code letter</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>code word</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>coding alphabet</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>common valuation</td>
<td>19</td>
<td>24</td>
</tr>
<tr>
<td>entropy</td>
<td>9</td>
<td>6</td>
</tr>
<tr>
<td>Huffman code</td>
<td>13</td>
<td>10</td>
</tr>
<tr>
<td>instantaneous code</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>length of a code word</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>minimal code</td>
<td>15</td>
<td>20</td>
</tr>
<tr>
<td>n-th extension of a source</td>
<td>10</td>
<td>7</td>
</tr>
<tr>
<td>optimum code</td>
<td>13</td>
<td>10</td>
</tr>
<tr>
<td>original outer code</td>
<td>17</td>
<td>21</td>
</tr>
<tr>
<td>prefix</td>
<td>6</td>
<td>2</td>
</tr>
<tr>
<td>reduced outer code</td>
<td>17</td>
<td>21</td>
</tr>
<tr>
<td>Shannon code</td>
<td>12</td>
<td>9</td>
</tr>
<tr>
<td>uniquely decodable code</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>valuation</td>
<td>16</td>
<td>21</td>
</tr>
<tr>
<td>word</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>word list</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>zero-memory source</td>
<td>8</td>
<td>6</td>
</tr>
<tr>
<td>$\mu$-blocks</td>
<td>18</td>
<td>24</td>
</tr>
<tr>
<td>$\mu$-th outer code</td>
<td>17</td>
<td>21</td>
</tr>
<tr>
<td>$R_I$</td>
<td>14</td>
<td>19</td>
</tr>
<tr>
<td>$S_I$</td>
<td>7</td>
<td>6</td>
</tr>
</tbody>
</table>
BIBLIOGRAPHY


ERRATA

Page 5, line 6T*: Replace second last "<" by "=".

Page 7, line 3T: Replace the line by: "of the information one receives if \( \alpha_i \) did occur. It also justifies...".

Page 7, line 8T: Replace "i = 1, 2, ..." by "j = 1, 2, ...".

Page 14, line 5B: Replace "=" by ">".

Page 20, line 6T: Replace the second sentence by:
"Theorem 3 states a sufficient condition for the two codes to be equivalent".

Page 20, line 7B: Replace "0 \( \leq \) b ..." by "0 < b ...".

Page 24, line 3T: Replace "Lemma 10" by "Lemma 11".

Page 25, line 7T: Replace the line by:
\[
(D - 1) \sum_{g=1}^{G-1} D^{-n_g} < R_{E_\mu} < (D - 1) \sum_{g=1}^{G} D^{-n_g}.
\]

Page 26, line 5T: Replace the equation by:
\[
(D - 1) \sum_{g=1}^{G} D^{-n_g} = R_{E_\mu}.
\]

Page 27, line 3B: Replace "dented" by "denoted".

* mT: n-th line from top

mB: m-th line from bottom
Page 29, line 12T: Replace the line by: "is a minimal code with $n_i^0 \leq n_i \leq n_i^1$ if the following sufficiency conditions hold:
(a) $n_i^0 \geq n_i - 1$, $i = 1, 2, \ldots, K$.
(b) If $n_i^0 < \tilde{n}_i^j$, $j = 1, 2, \ldots, J$, then
\[ v_{i,j} \geq v_{i,r}, \text{ for any } r \text{ such that } r > J. \]

Page 29, lines 5B to 1B: Replace these lines by:
"Theorem 3: Let $E = \{(p_i, n_i)\}^K_{i=1}$ be the Shannon code for a fixed set of probabilities $\{p_i\}^K_{i=1}$. Let $M = \{(p_i, n_i^0)\}^K_{i=1}$ be a minimal code satisfying conditions (a) and (b) of Theorem 2. Let $C = \{(p_i, \tilde{n}_i)\}^K_{i=1}$ be a Huffman encoding of $\{p_i\}^K_{i=1}$. Then for all $i$, $\tilde{n}_i \leq n_i$. That is, every minimal code formed by reducing only the maximum valuations is a Huffman code."

Page 34, line 7T: Under $p_1$, replace ".18" by ".15".

Page 43, line 8B: Under $p_1$, replace ".0164" by ".0175"

Page 49, line 6B: Delete: "By Theorem 3, $\tilde{n}_i \leq n_i$; hence $r_i \geq 0$, $i = 1, 2, \ldots, K$."