X. PROCESSING AND TRANSMISSION OF INFORMATION*

Prof. P. Elias
Prof. R. M. Fano
Prof. D. A. Huffman
Prof. C. E. Shannon
Dr. M. V. Cerrillo
Dr. M. P. Schützenberger

R. R. Capraro
E. Ferretti
J. V. Harrington
Cynthia H. Hsaio
K. Joannou
R. M. Lerner

R. S. Marcus
L. S. Onyshkevych
E. T. Schoen
S. H. Unger
J. M. Wozencraft
W. A. Youngblood

A. INFORMATION FLOW PATTERN IN A RECTANGULAR ARRAY OF CELLS

Given cells of the same structure in a rectangular array, and given that each cell is directly accessible (or connected) to a finite number of nearby cells, with what kinds of patterns of directly accessible cells can any cell be eventually accessible to every other cell in the array ?

By definition, "cell A is eventually accessible to cell B" means that a sequence of cells beginning with A and ending with B consists of only the cells that are directly accessible to the subsequent cells.

A necessary condition for any cell to be eventually accessible to every other cell in a rectangular array is that the number of directly accessible cells is three or more. A pattern of three will be sufficient if the following conditions are satisfied. Assume that any typical cell with coordinates (0, 0) is directly accessible to the cells with coordinates (a_1, a_2), (b_1, b_2), and (c_1, c_2). Then,

(1) The values of the determinants $\begin{vmatrix} a_1 & b_1 \\ a_2 & b_2 \end{vmatrix}$, $\begin{vmatrix} b_1 & c_1 \\ b_2 & c_2 \end{vmatrix}$, and $\begin{vmatrix} c_1 & a_1 \\ c_2 & a_2 \end{vmatrix}$ have to be of the same sign (and none of them may equal zero).

(2) The greatest common divisor of these determinants has to be unity.

Cynthia H. Hsiao

B. THEOREM ON SYMMETRIC SWITCHING FUNCTIONS

In the Quarterly Progress Report of April 15, 1956, page 78, a decimal procedure for identifying symmetric functions was described. The purpose of the present report is (a) to give a precise definition of self-dual boolean functions, (b) to supplement section 5 of the previous report by giving the necessary and sufficient conditions for a symmetric boolean function to have coindices of the form p/p, and (c) to prove that self-dual symmetric boolean functions with coindices of the form p/p do not exist.

THEOREM: The class of self-dual symmetric (1) boolean functions of n variables with coindices of the form p/p, where p is necessarily a positive integer not exceeding 2^{n-1}, is empty.

DEFINITION 1: A boolean function of n variables, $T(x_n \ldots x_k \ldots x_1)$, not necessarily symmetric, is called self-dual if and only if

*This research was supported in part by Purchase Order DDL-B187 with Lincoln Laboratory, which is supported by the Department of the Army, the Department of the Navy, and the Department of the Air Force under Contract AF19(122)-458 with M. I. T.

$$T(x_n \dots x_k \dots x_1) \equiv T'(x'_n \dots x'_k \dots x'_1)$$

Example for n = 3.

$$x_1x_2 + x_2x_3 + x_3x_1 \equiv (x_1 + x_2)(x_2 + x_3)(x_3 + x_1)$$

$$x_1x_2 + x_2x_3 + x_3x_1 \equiv (x_2 + x_1x_3)(x_3 + x_1)$$

$$x_1x_2 + x_2x_3 + x_3x_1 \equiv x_1x_2 + x_2x_3 + x_3x_1$$

DEFINITION 2a: The complement of the symmetric boolean function $S_{\{a_j\}}(x_n \dots x_k \dots x_1)$ will be denoted as $S'_{\{a_j\}}(x_n \dots x_k \dots x_1)$ and it obeys the identity $S'_{\{a_j\}}(x_n \dots x_k \dots x_1) \equiv S_{\{a'_j\}}(x_n \dots x_k \dots x_1)$, where $\{a_j\}$ denotes any sequence of the so-called a-numbers (1) of the symmetric function, and $\{a'_j\}$ denotes those members of the complete set of a-numbers $a_0, a_1, \dots, a_n$ not present in $\{a_j\}$, where $0 \leq j \leq n$ (j and n are integers).

DEFINITION 2b: A symmetric boolean function has coindices (2) of the form p/p if and only if $S_{\{a_j\}}(x_n \dots x_k \dots x_1) \equiv S_{\{a_j\}}(x'_n \dots x'_k \dots x'_1)$. [Coindices give a comparison of the number of ones to zeros in each of the n columns of a matrix corresponding to the n variables of the boolean function when each of the rows of this matrix is a term of the canonic sum form for the boolean function.]

Example of definition 2a.

$$S'_{2,4}(x_4x_3x_2x_1) \equiv S_{0,1,3}(x_4x_3x_2x_1)$$

Example of definition 2b.

$$S_{2,3}(x_5x_4x_3x_2x_1) \equiv S_{2,3}(x'_5x'_4x'_3x'_2x'_1)$$

[Here use is made of an identity not necessary in the proof of the theorem. The identity is: $S_{\{a_j\}}(x_n \dots x_k \dots x_1) \equiv S_{\{n-a_j\}}(x'_n \dots x'_k \dots x'_1)$.]

PROOF: Assume that a symmetric boolean function which is self-dual and has coindices of the form p/p exists, and show that this assumption is contradictory.

(a) For a symmetric boolean function to have coindices of the form p/p implies, by definition 2b, that

$$S_{\{a_j\}}(x_n \dots x_k \dots x_1) \equiv S_{\{a_j\}}(x'_n \dots x'_k \dots x'_1)$$

(b) To have a symmetric boolean function self-dual implies, by definition 1, that

$$S_{\{a_j\}}(x_n \dots x_k \dots x_1) \equiv S'_{\{a_j\}}(x'_n \dots x'_k \dots x'_1)$$

but, by definition 2a,

$$S'_{\{a_j\}}(x_n \ldots x_k \ldots x_1) \equiv S_{\{a'_j\}}(x_n \ldots x_k \ldots x_1)$$

hence,

$$S_{\{a_j\}}(x_n \ldots x_k \ldots x_1) \equiv S_{\{a'_j\}}(x'_n \ldots x'_k \ldots x'_1)$$

Thus, combining steps (a) and (b), we arrive at the contradiction that the expressions

$$S_{\{a_j\}}(x'_n \ldots x'_k \ldots x'_1) \quad \text{and} \quad S_{\{a'_j\}}(x'_n \ldots x'_k \ldots x'_1)$$

are to be identical, but,

$$S_{\{a_j\}}(x'_n \ldots x'_k \ldots x'_1) \not\equiv S_{\{a'_j\}}(x'_n \ldots x'_k \ldots x'_1)$$

since the left- and right-hand sides are complementary expression by definition 2a. Q. E. D.

A. A. Mullin

[Mr. Mullin is a member of the Switching Circuits Laboratory, M. I. T.]

References

1. C. E. Shannon, A symbolic analysis of relay and switching circuits, Trans. AIEE 57, 713-722 (1938).

2. E. J. McCluskey, Jr., Detection of group invariance or total symmetry of a boolean function, Bell System Tech. J. 35, 1445-53 (1956).

C. DISCRETE NOISELESS CODING

PART I. REVIEW OF PRESENT KNOWLEDGE OF THE SUBJECT

1. Summary

The problem of efficient coding (in the information theory sense) for finite, discrete, no-memory message sources and for finite, discrete, no-memory, noiseless channels is considered. Important known results and methods and some new results are described. Various classes of the coding problem are clearly distinguished. Emphasis is placed on the classes in which the number of message blocks is restricted either to the number of original messages or to the number of channel symbols, whichever is larger. However, procedures for larger numbers of message blocks, which lead to perfect efficiency, are also discussed. Various bounds on the efficiency are described for different procedures.

The case of cost-weighted channel symbols is discussed in parallel with the equal-cost case, which has received the most attention, thus far, in the literature of the subject. Cost-weighted symbols include those which have, for instance, unequal time durations. An extension of the Shannon procedure and bounds to this cost-weighted case is described. The interesting question of the admissibility of proper signal sets in the cost-weighted case is raised but not solved.

Details of the work presented in Part I are given in reference 1.

2. Introduction

2.1 The Communication System and Efficiency

Following Shannon (2), we describe the essential features of the communication system. We consider only discrete systems in which both the message and the signal are selected from sets of a finite number of elements. The message set contains m elements (written m_i; $i = 1, 2, \ldots, m$). The signal set contains D elements, called "channel symbols" (written d_j; $j = 1, 2, \ldots, D$). Coding may then be described as the process whereby the message m_i or a sequence of messages, called a "message block" (written M_k; $k = 1, 2, \ldots, M$), is replaced by a sequence of channel symbols called a "code word" and written W_k; $k = 1, 2, \ldots, M$.

In general, we associate with each symbol d_j a cost c_j. This cost is most frequently thought of as a time duration of the symbol ($c_j = t_j$ seconds), but it may be expressed in terms of power, bandwidth, or any other economically motivated consideration. If we do not wish to specify a particular unit, we give the general unit "unc" to the c_j. (Unc, which rhymes with bunk, is an abbreviation for "unit cost.")

Each message has a certain a priori probability of occurrence (written p_i). In this discussion, we consider that these probabilities stay fixed; the source is then called "no-memory."

The channel capacity, C, is defined as the maximum rate at which information may be transmitted over the channel. The rate of transmission, R, is given by

$$R = \frac{-\Sigma p_k \log p_k}{\Sigma p_k c_k} = \frac{H}{\overline{w}} \text{ bits/unc} \tag{1}$$

where p_k is the probability of M_k (a single message), c_k is the total cost of all the symbols in W_k in uncs, H is the entropy of the message source in bits/message, and $\overline{w}$ is the average cost of messages in uncs/message. The summations are taken over the entire range of the index. All logarithms are to base 2 unless otherwise noted.

The fundamental theorem for a noiseless channel states (2) that it is possible to encode any message source into any channel with capacity C in such a way that

$$R = C - \epsilon \quad \text{for any } \epsilon > 0 \tag{2}$$

and it is not possible to perform an encoding in such a way that $R > C$. One criterion for a good code, then, will be its efficiency, η, which is defined by $\eta = R/C$.

2.2 Classes of the Coding Problem

The coding problem is divided into three classes, according to the relative numbers of original messages, m, message blocks (and hence code words), M, and channel symbols, D. In Class I the number of messages is greater than or equal to the number of channel symbols. The original messages are represented directly by code words. In Class II there are fewer messages than channel symbols. Here the messages are coded first into message blocks which are then represented directly by channel symbols. In Class III, the most complicated type of coding, messages are coded into blocks which are then represented by code words. Summarizing, we find the following relations:

Class I	$m = M \geqslant D$
Class II	$m < M = D$
Class III	$m < M > D$

Each of these classes can be subdivided into two groups:

Group A	Equal-cost, all c_j are equal
Group B	Cost-weighted, some c_j may be unequal

Classes IA, IIIA, and to a lesser extent IIIB, have received the most attention thus far. Class III has usually been analyzed for the limiting case of perfect efficiency. Here we consider all classes.

2.3 The Tree Graph for Coding

It is convenient to think of all of the possible code words as branches on a tree-like structure. (See Fig. X-1.) From the root on the left D branches extend to the right, each one representing one of the channel symbols. The projection of the length of each branch on a horizontal axis is made proportional to the cost of the symbol. But it can be shown (1) that $c_j = (1/C)(-\log P_{mj})$, where the P_{mj} are the maximizing probabilities of the channel, i.e., those values for p_i that are such that $R = C$. Therefore, we may make the projected length of each branch equal to $(-\log P_{mj})$. From the right node of each branch another "fan" of D branches extends to the right. The D^2 branches in these D fans then represent all possible code words with just two symbols. By further extensions, code words of greater numbers of symbols are represented. The code word which any branch represents can be found as the sequence of channel symbols associated with the branches on the path from the root to the branch in question (including that branch). The horizontal distance from the root to the far node of any branch is the "normalized cost" (written q_k) of the word, i.e., the cost multiplied by C. It

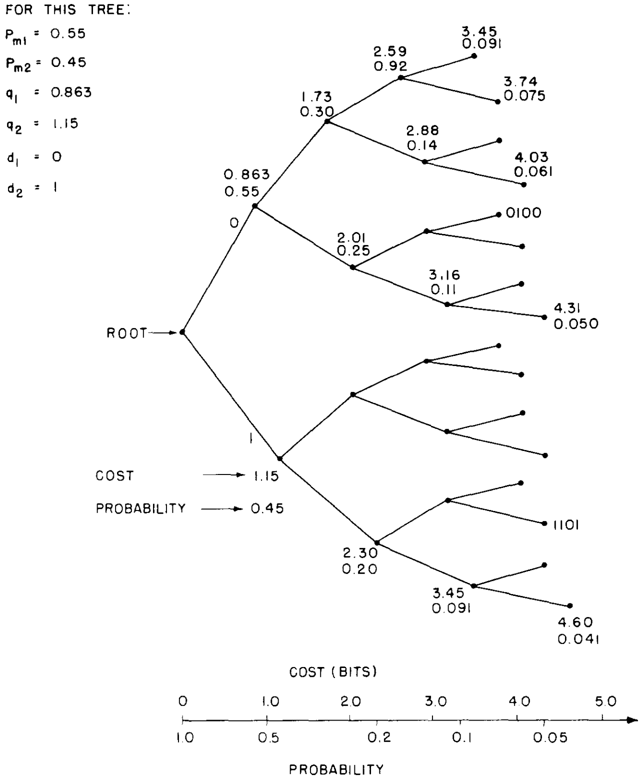

Fig. X-1. Tree graph for coding.

is measured in bits.

2.4 Decodability

One necessary restriction on the signal set (for error-free operation) is that it be decodable. That is, when a sequence of symbols arrives at the receiver we must be able to decompose this sequence in a unique manner into code words. A sufficient condition for this unique decomposability is that no code word be a prefix of any other code word.

We term a signal set which obeys the prefix rule as "proper" (3). But we may consider sets of code words of such a kind that any infinite sequence of channel symbols has at least one code word as a prefix. Such a set is called "complete."

3. Coding for Class I

3.1 The Shannon Procedure (Class IA)

Shannon (2) describes a procedure for picking the message set in the equal-cost case in such a manner that the number of symbols of each word is bounded as follows:

$$-\log_D p_k \leq n_k = [-\log_D p_k] < -\log_D p_k + 1 \tag{3}$$

where $[x]$ equals the smallest integer that is greater than or equal to x. This, in turn, gives the following bound on the average number of symbols per message ($\bar{n}$):

$$\bar{n} = \Sigma p_k n_k < \Sigma p_k \left(-\log_D p_k + 1\right) = \Sigma p_k - \frac{\Sigma p_k \log p_k}{\log D} = 1 + \frac{H}{\log D} \tag{4}$$

Hence,

$$\bar{q} = \bar{n} \log D < H + \log D \tag{5}$$

and

$$\eta = \frac{H}{\bar{q}} = \frac{H}{\bar{n} \log D} > \frac{H}{\frac{\log D}{H} + 1} \tag{6}$$

3.2 The Extended Shannon Procedure (Class IB)

Let us now consider how the Shannon procedure may be extended to the case of cost-weighted symbols. It can be shown (1, 2) that the costs to be associated ideally with the words designating the various messages are given by

$$q_k = -\log p_k$$

If this could be done for all k, we would get a perfect efficiency of $\eta = 1$.

It is true here, as in the equal-cost case, that $(-\log p_k)$ may not correspond to the cost of any possible word. However, we may describe the following extension to the Shannon procedure:

Order the messages according to decreasing probability. Draw a vertical line through the tree at a distance $(-\log p_1)$ from the root. This line will cut some branches of the tree which form a cut set. (Some of the branches may be cut at their far nodes.) For W_1 pick the cheapest word corresponding to any branch cut. Draw another line at distance $(-\log p_2)$. For W_2 pick the cheapest word, which is not prefixed by W_1, that corresponds to any branch cut. Continue this procedure, always picking the cheapest branch cut that is not prefixed by previously chosen words, until all M words are chosen.

For this manner of picking words, the lowest the cost will be is $q_k = -\log p_k$, which is a perfect match. But the cost can be no more than this value plus the normalized cost of the costliest symbol (i.e., the length of the longest branch), which is $(-\log P_{mD})$, which we write L_D. Thus the cost of each word is bounded as follows:

$$-\log p_k \leq q_k < -\log p_k + L_D \tag{7}$$

There is no possibility of equality on the right-hand side, since if a line passes through a node we always pick the branch on the left of the node. We can then bound the average cost as follows:

$$\bar{q} = \Sigma p_k q_k < \Sigma p_k (-\log p_k) + \Sigma p_k L_D = H + L_D \tag{8}$$

that is,

$$H \leqslant \bar{q} < H + L_D \tag{9}$$

Hence,

$$\eta = \frac{H}{\bar{q}} > \frac{H}{H + L_D} = \frac{1}{1 + \frac{L_D}{H}} \tag{10}$$

The proof that this procedure works is as follows: First, we note that the procedure would fail only if we were to "run out" of tree. However, an inspection of the method shows that if there are no more words to be chosen we have cut through the tree, i.e., the previous words chosen form a cut set. Now consider the structure function for the first p words, which is defined by

$$S_p = \sum_{k=1}^{p} p_k \tag{11}$$

Also consider the sum of the probabilities of all messages, which we write as E_{p+1}, where

$$E_{p+1} = \sum_{k=1}^{p} p_k \tag{12}$$

Since $q_k \geqslant -\log p_k$, we have

$$S_p = \sum_{k=1}^{p} 2^{-q_k} \leqslant \sum_{k=1}^{p} 2^{\log p_k} = \sum_{k=1}^{p} p_k = E_{p+1} \tag{13}$$

But for a cut set of p words it can easily be shown (1) that $S_p = 1$. Hence, $E_{p+1} = 1$. This indicates that if the tree ever is used up (i.e., a cut set were formed) then all the messages would have also been used up. Therefore, we can never have a message for which there is no code word; and the proof is completed.

3.3 Proof of the Fundamental Theorem by the Use of the Extended Shannon Procedure

Let us show how the bound established on η can be applied to "block coding" (Class III coding). Consider equal length block coding in which the message set consists of all possible permutations of length L of the messages. Thus $M = m^L$. The entropy of the message set, H, is given by $H = LH_o$, where H_o is the entropy of the original message source with m messages. The extended Shannon procedure gives us the bound:

$$\bar{q} < H + L_D = LH_o + L_D \tag{14}$$

Hence,

$$\eta > \frac{H}{H + L_D} = \frac{LH_o}{LH_o + L_D} = \frac{1}{1 + \frac{L_D}{H}} \tag{15}$$

Thus the efficiency approaches one as L approaches infinity. This is a direct constructive proof of the fundamental theorem for the cost-weighted channel. It complements Shannon's proof (2) for the binary equal-cost channel.

4. Optimum Codes for Class I

For Class IA, the Huffman procedure (4) gives us a relatively simple algorithm for obtaining the optimum code, i.e., the signal set which gives the maximum efficiency. There seems to be no simple extension to the cost-weighted case.

The special case of equiprobable messages $\left(p_k = \frac{1}{M}\right)$ is solved, however. First, we define a "vertical signal set" as a set of branches cut by a vertical line drawn at some cost-distance, y, from the root of the tree. This line may cut a branch at either end. Then we assert that an optimum signal set for the channel is a vertical cut set with M branches. For the D-symbol channel, the optimum signal set consists of the M least costly words from some vertical set. This vertical set is found by a relatively simple procedure (1).

5. Proper and Nonproper Coding

5.1 Sardinas-Patterson Rule for Unique Decomposability

We have seen that a sufficient condition on the signal set for the code to be uniquely decomposable is that it obey the prefix rule. That this is not a necessary condition may be seen by considering the signal set $W_1 = 0$, $W_2 = 01$. Sardinas and Patterson (5) give a procedure for determining whether or not a given signal set is uniquely decomposable. This procedure is as follows.

The signal set itself is called "segment class zero," written Seg_o. If one word prefixes another, the remainder of the second word is in segment class one, written

Seg_1. If one word in Seg_o prefixes a word in Seg_1, or vice versa, the remainder of the longer word is placed in Seg_2. In like fashion Seg_{i+1} is generated from Seg_i and Seg_o. The rule then states that the code is uniquely decomposable if no word in Seg_o appears in Seg_i, $i \geq 1$.

Since the maximum word length is bounded (n_{max}), the algorithm will give a definite answer in a finite number of steps. There are two possibilities for success: (a) Seg_i is empty for some i, and hence Seg_j is empty for all $j \geq i$; (b) $\mathrm{Seg}_j = \mathrm{Seg}_i$ (Seg_i is not empty) for some $j > i$. For the second case, the segment classes repeat in a periodic fashion, the period being j-i. In case (a) the code has "local decodability" (a term invented by Schützenberger), in case (b) it does not. A code has local decodability if there exists L with the property that for any given m we can uniquely decode the first $m - n_{max}$ symbols into messages, once the first L symbols are known.

5.2 An Inequality for Proper Signal Sets

We shall now find necessary and sufficient conditions on the set of words in a signal set so that there exists a proper signal set with the given set of costs. We do this, first, for D = 2. The cost of a word is characterized by the number of d_1 and d_2 symbols in it. Let the number of code words with x d_1's and y d_2's be given by N(x, y). The total number of words with this composition is $C_{x,y}$:

$$C_{x,y} = \frac{(x+y)!}{x!\,y!} \tag{16}$$

Hence,

$$N(x,y) \leq C_{x,y} \tag{17}$$

But if any code word with composition (x', y') is used in such a manner that $x' \leq x$ and $y' \leq y$, then we cannot use all $C_{x,y}$ words of composition (x, y) as code words. Specifically, there are $C_{x-x', y-y'}$ words of composition (x, y) which are prefixed by the word of composition (x', y'); therefore none of the former can be used simultaneously with the latter. Similarly, if we have N(x'y') words of composition (x', y'), we cannot use $N(x',y')C_{x-x', y-y'}$ words of composition (x, y). In the same way, if there are any words of composition (x", y") with the property that $x" \leq x$ and $y" \leq y$, we cannot use $N(x",y")C_{x-x", y-y"}$ additional words of composition (x, y). A necessary condition on the N(x, y) for a prefix code is then given by

$$\sum_{i=0}^{x} \sum_{j=0}^{y} N(x-i, y-j)\, C_{i,j} \leq C_{x,y} \tag{18}$$

Conditions 18 must hold simultaneously for all (x, y) compositions. It should be clear that if it holds for some (x, y) it also holds for any (x', y') that are such that $x' \leqslant x$ and $y' \leqslant y$. Also, if it holds for some (x, y) it also holds for any (x+i, y+j), as long as $N(x+i, y+j) = 0$ for all $i + j > 0$. Then if, for a given signal set, the maximum $n = x + y$ for a code word is n_{max}, conditions 18 reduce to the $n_{max} + 1$ inequalities that are such that $x + y = n_{max}$.

We note that, if the equality holds for some (x, y), the tree is full for that (x, y). If the equality holds for all (x, y) that are such that $x + y = n$, then the tree is full and the signal set is a cut set. We may see in the following way that conditions 18 are sufficient. Make code words for all (x, y) with the property that $n = x + y = 1$. Then make code words obeying the prefix condition for all (x, y) with the property that $n = x + y = 2$. Continue this procedure. If it fails, at least one of conditions 18 must not be satisfied.

We can easily extend these results to the D-symbol case (1).

5.3 Admissibility of Proper Signal Sets

Could any of these nonprefix codes be more efficient than the best prefix code? This is answered (6, 7) in the negative in the equal-cost case by some simple arguments. We may say, then, that the class of prefix codes is admissible when we are considering efficiency for the equal-cost case. It is believed that the same result should hold for the cost-weighted case, but this has not been proved.

We could prove the hypothesis for the cost-weighted case if we could prove a somewhat stronger hypothesis which is probably true: Any signal set that passes the Sardinas-Patterson test can be changed into a signal set that is proper, if it is not already so, merely by rearranging some of the symbols in the words. This proper signal set would then have the same cost structure as the first. We could also prove the hypothesis by proving the following statement, which is also probably true: If one of conditions 18 is not satisfied, the signal set fails the Sardinas-Patterson test.

6. Other Results

6.1 Coding for Class II

We conjecture that the optimum solution for Class IIA is that the message set should be the vertical cut set with D elements. For this solution, we have the bound

$$\eta > 1 - \frac{s}{\log D} \tag{19}$$

where $s = -\log p_m$ = the self-information of the least probable original message, i.e., the length of the longest branch on the message tree.

Although the message set must be complete in Class II, it need not be proper. It is shown that for some cases a nonproper message set is more efficient than any proper one. However, coding for the nonproper message set is somewhat more complicated.

For Class IIB, a procedure is demonstrated which gives the following bound:

$$\eta > 1 - \frac{s}{H - D} \qquad (20)$$

where H is the entropy of the message set given by the procedure.

6.2 Coding for Class III

The block coding scheme known as "balanced coding" in which one attempts to code a message set of equal or nearly equal probabilities for the message blocks into a signal set of equicostly (or nearly so) signal words is described. Specifically, pick a given M. Then take the first vertical cut set in both the message and signal trees which has M members. The message blocks in order of decreasing probability are then matched to the signal words in order of increasing cost. Using this procedure, we obtain the bound

$$\eta > \frac{1}{1 + \frac{L_D}{H}} > \frac{1}{1 + \frac{L_D}{\log M - s}} \qquad (21)$$

It is shown that balanced coding is better than equal-length block coding (for a given M) in three respects: (a) the costliest signal word is less expensive; (b) the bound for η is better; (c) for a class of message probability distributions with one large probability close to one, balanced coding gives the best possible efficiency for a small M.

As in Class II, a nonproper message set can be used to achieve greater efficiency than any proper message set for certain cases.

6.3 Remarks

The thesis (1) also contains discussions of bounds on the efficiency in Classes IA and IB; a geometric picture of the coding problem due to Mandelbrot (8); examples of codes for the letters of the English language; some other coding procedures of Huffman, Fano, and Blachman; and a list of some more general types of coding problems than those that have been mentioned.

PART II. A RELATION FOR OPTIMUM CODING

The Shannon procedure shows that we can find a code with the property that $n_k < -\log_D p_k + 1 = I_{dk} + 1$ for all words, W_k, where I_{dk} is the self-information of the k-th message to the base D which equals $-\log_D p_k$. It may be suspected that for the optimum code n_k will not be much greater than I_{dk} for any k. We show here that this is true in a ratio sense but not in an absolute sense. That is, it is true that for large I_{dk}

$$n_k^* < rI_{Dk} \tag{1}$$

where n_k^* is n_k for the optimum code and r is a parameter which varies with D. For $D = 2$, $r \doteq 1.44$, and for $D \to \infty$, $r \to 2$. However, we can demonstrate probability distributions with the property that $n_k \doteq rI_{Dk}$, which holds for large I_{Dk} for at least one k, say $k = \alpha$. Thus $n_\alpha^* - I_{Dk} > b$, where b may be made arbitrarily large.

To prove these assertions we make use of the Huffman procedure (4) for finding the optimum signal set. Consider a message, M_α, with probability S_o. The number of symbols, n_α^*, in W_α for the optimum code will be the number of times S_o is combined with other probability sums in the tree-like structure of the Huffman method. See Fig. X-2.

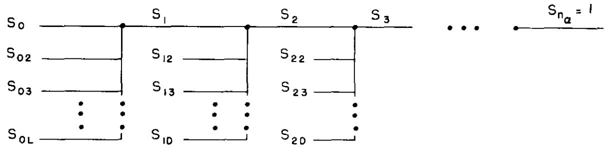

Fig. X-2. Pertaining to the construction of an optimum code.

To make n_α as large as possible, we make the intermediate sums $S_1, S_2, S_3, \ldots$ as small as possible. We do this, in turn, by making as small as possible the probabilities $S_{02}, S_{03}, \ldots, S_{0L}$ $(2 \leqslant L \leqslant D)$, and the sums $S_{12}, S_{13}, \ldots, S_{1D}$, $S_{22}, S_{23}, \ldots, S_{2D}$, $S_{32}, \ldots$. Thus,

$$S_1 = S_0 + S_{02} + S_{03} + \ldots + S_{0L} > S_0 \tag{2}$$

$$S_2 = S_1 + S_{12} + S_{13} + \ldots + S_{1D} \tag{3}$$

But S_{12}, S_{13}, and so on, are all $\geqslant S_o$ (otherwise they would have been combined with S_{02}, S_{03}, and so on, instead of S_o). Hence,

$$S_2 \geqslant S_1 + (D-1)\, S_o > D\, S_o \tag{4}$$

Thus, if $S_o \geqslant \frac{1}{D}$, then $n_\alpha < 2$. Similarly,

$$S_3 = S_2 + S_{22} + S_{23} + \ldots + S_{2D} \tag{5}$$

But S_{22}, S_{23}, and so on, $\geqslant S_1$. Hence,

$$S_3 > D\, S_o + (D-1)\, S_o = (2D - 1)\, S_o \tag{6}$$

Thus, if $S_o \geqslant (2D - 1)^{-1}$, then $n_\alpha < 3$. But, from the construction, we see that we can find a p_k set that is such that, if $S_o = (2D - 1)^{-1} - \epsilon$, $\epsilon > 0$, then $n_\alpha^* = 3$.

In general, we find that

$$S_j \geqslant S_{j-1} + (D-1)\, S_{j-2} > f_j\, S_o \tag{7}$$

where

$$f_j = f_{j-1} + (D-1)\, f_{j-2} \tag{8}$$

The solution to difference equation 8 is of the form

$$f_j = A\, \beta_1^j + B\, \beta_2^j \tag{9}$$

for which we find

$$\begin{aligned} \beta_1 &= \frac{1}{2} + \frac{1}{2}(4D - 3)^{1/2} \\ \beta_2 &= \frac{1}{2} - \frac{1}{2}(4D - 3)^{1/2} \end{aligned} \tag{10}$$

Thus, for $S_o \geqslant f_j^{-1}$, $n_\alpha^* < j$. Because of the exponential nature of f_j we can make the following approximations for large j:

$$f_j \doteq A\, \beta_1^j \tag{11}$$

Hence, for $S_o > A\, \beta_1^j$, $n_\alpha^* < j$.

Now suppose $S_o = A\, \beta_1^{-j+\delta}$, where $0 \leqslant \delta < 1$. Then $S_o \geqslant A\, \beta_1^{-j}$ and $n_\alpha < j$. But $I_{D\alpha} = -\log_D S_o \doteq j \log_D \beta$. Hence,

$$n_\alpha < \frac{I_{D\alpha}}{\log_D \beta_1} = rI_{D\alpha} \tag{12}$$

where $r = \dfrac{1}{\log_D \beta_1}$. But, as before, we know we can construct a set of p_k that is such that $n_\alpha \doteq I_{D\alpha}$. For D = 2, $\beta_1 = \frac{1}{2} + \frac{1}{2}\sqrt{5} \doteq 1.618$, $r \doteq \dfrac{1}{\log_2 1.618} \doteq 1.44$. For $D \to \infty$, $\beta_1 \to \frac{1}{2}\sqrt{4D} = D$, $r \to \dfrac{1}{\log_D D^{1/2}} = 2$.

This completes the proof. Since there is no known simple extension of the Huffman procedure to the cost-weighted case, these results are not directly extendable to that case.

Other relationships illustrating the fact that code words do not have to be too expensive have been found (1). From Eq. 7 of Part I we know, for the Shannon code, that

$q_k < -\log p_k + L_D$. Hence,

$$S = \Sigma 2^{-q_k} > \Sigma p_k 2^{-L_D} = P_{mD} \tag{13}$$

Together with the original Kraft inequality (1), we then have

$$P_{mD} < \Sigma 2^{-q_k} \leqslant 1 \tag{14}$$

For the equal-cost case, this reduces to

$$\frac{1}{D} < \Sigma D^{-n_k} = \Sigma 2^{-q_k} \leqslant 1 \tag{15}$$

It can be shown (1) that relation 15 also holds for the optimum code in the cost-weighted case; that is,

$$S = \Sigma 2^{-q_k} > \frac{1}{D} \tag{16}$$

R. S. Marcus

References

1. R. S. Marcus, Discrete noiseless coding, S. M. Thesis, Department of Electrical Engineering, M. I. T., January 1957.
2. C. E. Shannon, The mathematical theory of communication, Bell System Tech. J. 27, 379 (1948).
3. A. E. Laemmel and J. M. Brogan, Coded transmission of information, Research Report R-325-53, PIB-261, Microwave Research Institute, Polytechnic Institute of Brooklyn, January 1954.
4. D. A. Huffman, A method for the construction of minimum-redundancy codes, Proc. IRE 40, 1098 (1952).
5. A. A. Sardinas and G. W. Patterson, A necessary and sufficient condition for the unique decomposition of coded messages, IRE Convention Record, Part 8, 1953.
6. B. Mandelbrot, On recurrent noise limiting coding, Proc. Symposium on Information Networks, Polytechnic Institute of Brooklyn, 1954.
7. B. McMillan, Two inequalities implied by unique decipherability, Trans. IRE, PGIT, vol. IT-2, no. 4 (Dec. 1956).
8. B. Mandelbrot, Théorie des informations en l'absence de bruit, Institut de Statistique, University of Paris, 1955.