

**NASA TECHNICAL
REPORT**



NASA TR R-261

c/

NASA TR R-261

LOAN LIBRARY
APRIL 1968
KIRTLAND AFB,



**DATA COMPRESSION WITH
ERROR-CONTROL CODING
FOR SPACE TELEMETRY**

by Thomas J. Lynch

*Goddard Space Flight Center
Greenbelt, Md.*

CORRECTION FOR PAGE 61

of

NASA TECHNICAL REPORT NASA TR R-261
(June 1967)

by

THOMAS J. LYNCH
Goddard Space Flight Center
Greenbelt, Maryland

*Completed
15 June 68 Jf*

Replace material from the word "Then" in the middle of the page to the footnote with the following:

$$HH'x = \left[\begin{array}{c|c} - & \frac{I}{B_1 + B_2} \\ \hline \frac{I}{B_1 + B_2} & - \\ \hline - & \frac{O}{A_2 B_2} \\ \hline \frac{O}{A_2 B_2} & - \end{array} \right] \left[\begin{array}{c} -x_1 \\ 0 \end{array} \right] = \left[\begin{array}{c} -x_1 \\ 0 \end{array} \right]$$

leaving

$$A_2 B_2 = I,$$

$$B_2 = I,$$

$$B_1 + B_2 A_1 = 0$$

$$B_1 = -A_1 = A_1,$$

then

$$H = H'$$

TECH LIBRARY KAFB, NM



0068371

NASA TR R-261

DATA COMPRESSION WITH ERROR-CONTROL
CODING FOR SPACE TELEMETRY

By Thomas J. Lynch

Goddard Space Flight Center
Greenbelt, Md.

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

For sale by the Clearinghouse for Federal Scientific and Technical Information
Springfield, Virginia 22151 - CFSTI price \$3.00

ABSTRACT

The considerations involved in combining data compression and error control coding in space telemetry are analyzed through the use of two performance measures, D and R , which are similar to those measures used by Shannon for his *rate distortion function*. The average distortion D is a function of the source probability distribution, the overall system transitional probability matrix, and a cost matrix which signifies the relative importance of different types of data errors. The rate-ratio R is the reciprocal of the overall system compression ratio and includes the data expansion effect of additional timing data, identification data, and coding redundancy.

Different schemes for supplying the timing information in a compressed system are analyzed and compared. A new timing scheme is developed which requires, on the average, fewer time words for a large class of data sources. A method is developed for uniquely encoding and decoding an entire sequence of time words for compressed data, utilizing the strict monotonicity of the sequence.

The effects of the following system parameters and properties on the overall distortion and rate-ratio are analyzed: the error-control usefulness of natural data redundancy, errors in time information, the error-control usefulness of time word monotonicity, the probability distribution of the source, the bit-error probability of a binary symmetric channel, and the word compression ratio.

A rationale for comparing and choosing among three systems—uncompressed-uncoded, compressed-uncoded, and compressed-coded—is given in terms of the performance measures D and R .

CONTENTS

Abstract	ii
List of Symbols	iv
Chapter I. INTRODUCTION	1
Chapter II. THEORETICAL BACKGROUND	3
Telemetry	3
Data Compression	4
Coding	8
Rate Distortion	10
Chapter III. EFFECTS OF DATA COMPRESSION ON SOURCES	12
System Arrangement of Compressors	12
Entropy-Reducing Compression Effects	13
Information-Preserving Compression Effects	14
Chapter IV. THE TIME ENCODING PROBLEM	19
Nature of the Problem	19
Modified Predictor Algorithms	20
Three Basic Methods	22
Comparison of the Three Methods	29
Chapter V. TRADEOFF MEASURES	33
Measures Based on the Rate Distortion Concept	33
Rate-Ratios and Distortions for Three Systems	34
Effect of Time Errors on Distortion	39
Chapter VI. USE OF THE TRADEOFF MEASURES	43
A Rationale for Choosing a System	43
Effect of the Channel—An Example	45
Effect of the Compression Ratio	46
Effect of the Source Probability Distribution	48
Graphical Representation of Tradeoffs—An Example	52
Chapter VII. CONCLUSIONS	56
References	57
Appendix A—Coding Matrices and Error Probabilities for the Hamming (13,9) Code	59
Appendix B—Example of the Minimization of a Rate Distortion Function	65
Appendix C—Examples of Entropy-Reducing Compression Effects on First-Order Probability	73

LIST OF SYMBOLS

k	number of bits in telemetry data word
p	binary-symmetric channel bit-error probability
$\{x\}$	sampled quantized data source
x_i	data sample value
t_i	sample time
Δt	sample time interval
M_d	number of data levels
\hat{x}_t	predicted sample value
λ	tolerance in predictor algorithms
\bar{n}_T	theoretical average word length
\bar{n}_A	actual average word length
$H(X)$	entropy of the source x in bits
A	number of symbols in encoding alphabet
H	entropy in bits/sec
C	channel capacity in bits/sec
ϵ	arbitrarily small number
n	length of code word in bits
d	Hamming distance
d_{\min}	minimum distance
e_d	number of errors to detect
e_c	number of errors to correct
$P(x_j/x_i)$	conditional probability of going from x_i to x_j
M	data <i>source</i> set
Z	data <i>user</i> set

LIST OF SYMBOLS (Cont'd)

m_i	a source data word
r_i	a user data word
$c(m_i, z_j)$	cost of sending m_i and receiving z_j
$D(M, Z)$	distortion measure matrix
D	average distortion
$p(M)$	first-order probability of source M
$p(Z/M)$	transitional probability matrix for the transmission of m_i to z_j
$r(D)$	rate distortion function
$I(M; Z)$	average mutual information between M and Z
$H(M)$	entropy of source M
V	number of discrete levels in source for ER compression
$H(X)$	entropy of discrete source X
$H(Y)$	entropy of discrete source Y
$H(Y/X)$	conditional entropy of Y (given X)
$H(X/Y)$	conditional entropy of X (given Y)
$p(x_i)$	first-order probability of i^{th} level of x
R	redundancy
L	number of samples in a sequence
N_c	number of compressible sequences
n	number of words in a minor frame
M	number of minor frames in a major frame
S	number of data sources multiplexed in a major frame
$(CR)_m$	word compression ratio
L_m	number of compressible words left in main frame after compression
$(\overline{CR})_m$	average word compression ratio

LIST OF SYMBOLS (Cont'd)

$\overline{(\text{CR})}_{ma}$	actual average compression ratio
k_t	number of bits in time or location word
K	factor used in compression ratio formula for modified compression algorithm
m	number of sample time intervals between periodic timing words
q	number of incremental time words out of $m - 1$ such words to sequence encode
N_T	total number of sequences out of $m - 1$ incremental time words
N_q	total number of sequences of q incremental time words out of $m - 1$ such words
$(\text{CR})_w$	word compression ratio for sequence encoding
$a_{i,j}$	element in sequence coding matrix
\bar{c}	average number of compressible words per minor frame
\bar{q}	average number of compressible words sent per minor frame
UU	uncompressed, uncoded
CU	compressed, uncoded
CC	compressed, coded
R	rate-ratio: bits/uncompressed bits
K_c	factor, greater than unity, which accounts for error-control coding data expansion
R_{uu}	rate-ratio for UU
R_{cu}	rate-ratio for CU
R_{cc}	rate-ratio for CC
D_{uu}	average distortion for UU due to data errors only
$P(M)_{uu}$	$P(M)$ for UU
$P(Z/M)_{uu}$	$P(Z/M)$ for UU—data errors only
D_{cu}	average distortion for CU due to data errors only
$P(M)_{cu}$	$P(M)$ for CU

LIST OF SYMBOLS (Cont'd)

$(Z/M)_{cu}$	$P(Z/M)$ for CU—data errors only
d_{cc}	average distortion for CC due to data errors only
$(Z/M)_{cc}$	$P(Z/M)$ for CC—data errors only
n_c	number of errors corrected by a code
p_c	probability of correct decoding
T_1, T_2	absolute time words that are sent periodically
t_k	times of non-redundant data samples
t_e	erroneous value of t_k
(t_r/t_s)	time-word transitional probability matrix for receiving t_r , given that t_s is sent
$(Z/M)_{tcu}$	$P(Z/M)$ for CU—time errors only
d_{tcu}	average distortion for CU due to time errors only
l_t	coded-time-word length in bits
$(Z/M)_{tcc}$	$P(Z/M)$ for CC—time errors only
d_{tcc}	average distortion for CC due to time errors only
d_{suu}	average distortion for UU due to data and time errors
d_{scu}	average distortion for CU due to data and time errors
d_{scc}	average distortion for CC due to data and time errors
d_{max}	maximum allowable distortion

DATA COMPRESSION WITH ERROR-CONTROL CODING FOR SPACE TELEMETRY*

by

Thomas J. Lynch

Goddard Space Flight Center

Chapter I

INTRODUCTION

Among the sources of the current interest in the application of data compression techniques to space telemetry are increasing data rates, increasing transmission distances, and the need for more real-time data relay over ground links. Compression techniques have been used in audio (Reference 1) and television (Reference 2) transmission, and in each case the desired result was to reduce the bandwidth required to communicate satisfactorily. In the space telemetry situation bandwidth reduction and data volume reduction are both desired. Scientific satellites often have lifetimes of a year or more and can transmit data at rates of 100,000 bits/sec with almost continuous coverage from earth tracking and receiving stations (Reference 3). Data compression makes it possible to transmit data from more on-board experiments at these high rates by removing the redundancy inherent in many experiment outputs.

The application of compression techniques to space telemetry brings up the question of error control in the entire telemetry system, since much of the redundancy which once aided the experimenter in identifying obvious data errors would now be removed by the compression process. There is the possibility of putting back redundancy in the form of coding for error control, and herein lies the basic question: Is data compression worthwhile, since it requires coding, and if it is, what are the best combinations of compressors and encoders? To approach this question, one must study the tradeoffs between natural redundancy in an uncompressed data system and controlled redundancy in a compressed and coded data system; and such a study requires a quantitative measure of performance to apply to each system. The development of this measure, including a study of the effect of data compression on the data source and the problem of time encoding and time errors, is the purpose of this report.

The compression of electrical signals started in 1939 with the invention of the vocoder by Dudley (Reference 4). This device divided speech into separate frequency bands on the sending end,

*In its original form, this paper was submitted to the University of Maryland in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Electrical Engineering. The author wishes to express his appreciation to Professor Alan B. Marcovitz for his advice and guidance in this thesis.

and synthesized the speech on the receiving end by using a corresponding set of frequency bands for amplification. This technique has been improved and refined through the years, and is still in use (Reference 1). Various level-quantization techniques have been tried in television compression as well as variations on run-length encoding (Reference 2). In both audio and video compression, advantage was taken of the peculiar characteristics (typically non-linear) of human hearing and vision.

The application of data compression techniques to telemetry has received noticeable attention in the literature during the past five years (References 5, 6, 7, 8, 9). Some authors have considered coding along with the compression—for example, Shannon-Fano (noiseless) types (Reference 5) and run length types (Reference 7) of coding—but not for the purpose of error control.

The area of channel encoding and decoding to control errors introduced in transmission has been well covered in the literature. But source encoding, other than the Shannon-Fano (Reference 10), Huffman (Reference 11), or run-length types, has not been treated as exhaustively as its channel counterpart. Shannon's 1960 paper entitled "Coding Theorems for a Discrete Source with a Fidelity Criterion" (Reference 12), treats the questions of source encoding and channel encoding together. In that paper, Shannon looks at the entire communication system from source to recipient and considers errors or fidelity in such a way as to include the interactions of the source and channel.

The aim of this research is to study the considerations involved in combining error-control coding with data compression in a space telemetry system. A measure of performance has to be found which is consistent with data compression and coding.

The statistics of the original data source will be included in this measure. The requirement for more timing information with *compressed* as compared to *uncompressed* data will be included and various schemes for providing this timing information, particularly in a time-shared telemetry system, will be studied and compared. The effect of errors in this timing information on the final reconstructed data will also be included. In order to compare various combinations of compression and coding to the uncompressed and uncoded case, a measure of the error-control value of uncontrolled, or natural, data redundancy must be found.

Chapter II

THEORETICAL BACKGROUND

Particular features of some well-established topics will be outlined here for later reference. These topics are telemetry, data compression, coding, and rate distortion. A great deal has been written about telemetry and coding, but data compression and rate distortion are still relatively new topics with a correspondingly small literature. For these reasons, the coverage here in this chapter will be brief, and more discussion of data compression and rate distortion will follow in later sections.

Telemetry

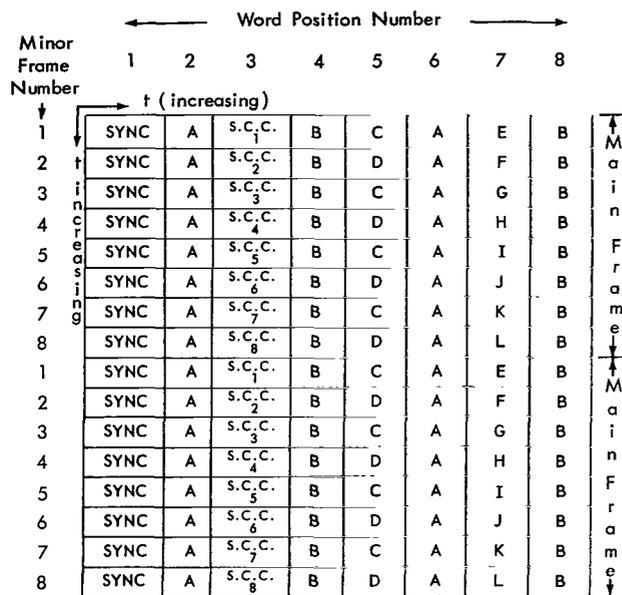
Telemetry From a Communication Standpoint

For the purposes of the present research, the telemetry system is taken as a time-sampled digital system with a quantization precision set by a k -bit binary code. The data source is assumed to be some analog function of time which is sampled at a constant rate. The digital space-to-ground link can be represented by a binary symmetric channel, wherein the probability of an error ($0 \rightarrow 1$, $1 \rightarrow 0$) is p and the probability of correct transmission ($0 \rightarrow 0$, $1 \rightarrow 1$) is therefore $1 - p$. At the ground data processing terminal, these digitized samples are converted to smooth time curves or are left in digital form for further analysis by digital computer.

The Time-Multiplexed Telemetry System

In a time-multiplexed telemetry system, time-shared sampling of a number of different sources is implemented by means of a definite order of intermixed sampling called *commutation*. The *commutator* may be mechanical or electronic, and its operation may be either inflexible (according to a predetermined pattern), or changeable (according to different patterns, each brought into use by ground command). The output of a commutator is then a sequence of samples from different sources, the pattern of samples repeating in some time period which is typically large in comparison to the time period between samples.

After the reception of such a time-multiplexed telemetry signal, a necessary operation in processing the data is to collect together the samples from the same source out of the time-multiplexed sequence. This *decommutation* depends for reliable operation upon *word-synchronization* in the multiplexed word sequence. This word-synchronization takes the form of an easily recognizable "sync" word that is inserted in the sampling sequence once every period or fractional period of the multiplexed sequence. The details of the multiplexed pattern with its included word synchronization can be illustrated (Figure 1).



A, B, C, D, E, F, G, H, I, J, K, L— Sampled Sources
 SYNC ————— Synchronization Word
 S.C.C. ————— Subcommutator Count

Figure 1—A time-multiplexed telemetry pattern.

Sources E through L are also sub-commutated, and since their subcommutation pattern has the longest period, this pattern determines the sub-commutator count.

As was mentioned above, an on-board clock might constitute one of the sources E through L and thereby produce a time readout once per main frame: thus an on-board clock is sometimes referred to as a *main-frame count*. Whether or not an on-board clock is included in the format, the *ground* time at the receiving station is recorded along with the received telemetry words, and in subsequent data processing this ground time is inserted into the time multiplexed pattern, typically in place of one or more of the sync words in a main frame, *after* they have been used to establish word synchronization.

In decommutation and subsequent data processing, it is only necessary to keep track of the absolute time as a reference point somewhere within a main frame, and the absolute times of all samples can then be obtained by keeping track of their *position* in the main frame.

Data Compression

There are basically two types of data compression (Reference 5): Entropy Reducing (ER) and Information Preserving (IP). An entropy-reducing (ER) data compression operation is an *irreversible* operation on the data source which results in an "acceptable" reduction in data fidelity. Examples

*Actual space telemetry main frames are typically larger than the one shown in Figure 1. A typical range of frame sizes is from 16 x 16 to 128 x 128. In addition, the frame is not always square.

of ER data compression are narrow-band filtering, logarithmic amplification, limiting, and statistical moment estimation. An information-preserving (IP) data compression operation is one that, when applied to the output of a data source, reduces the amount of energy required to transmit characteristics of the source, without reducing the fidelity of the source output. These characteristics can be reconstructed with a finite allowable error, and for this reason, this type of data compression is often called reversible. The two basic types of information-preserving data compression techniques for time-sampled data are *polynomial curve fitting* (Reference 9) and *statistical prediction* (Reference 6).

Information Preserving Data Compression

The following is a more detailed description of IP compression, particularly the polynomial curve-fitting type. This type of IP compression has received recent attention in the literature (Reference 9) and has been proposed for use in a space telemetry system (Reference 13).

An Information-Preserving data compression operation reduces the number of samples that need be transmitted in order to reconstruct the original waveform. This can be expressed mathematically as follows:

Consider a sampled, quantized data source $\{x\}$ such that

$$\{x\} = \{x_0(t_0), x_1(t_0 + \Delta t), x_2(t_0 + 2\Delta t), \dots, x_i(t_0 + i\Delta t) \dots\}$$

where x is quantized, i.e. each value of x can take on only one of M_d levels.

Another way of expressing the source to show the importance of sample times is:

$$\{x\} = \{x_0, t_0; x_1, t_1; x_2, t_2; \dots; x_i, t_i; \dots\}$$

In an uncompressed system, sample times need only be sent occasionally to maintain synchronization. When these samples are put through an Information-Preserving data compressor, some do not appear at the output. These samples can be later reinserted into the received data stream according to a reconstruction algorithm that complements the compression algorithm. An important problem in an Information-Preserving type of data compression is that not only must sampled values be sent,* but indications of *times of occurrence* of these sampled values must also be sent.

Polynomial Predictors

In this technique the next sample is predicted to lie on an n^{th} order polynomial, as defined by $(n + 1)$ previous samples (Reference 9). Mathematically:

$$\hat{x}_t = x_{t-1} + \Delta x_{t-1} + \Delta^2 x_{t-1} + \dots + \Delta^n x_{t-1},$$

*An examination of the time encoding problem is presented in Chapter IV.

where

$$\begin{aligned}\Delta x_{t-1} &= x_{t-1} - x_{t-2} , \\ \Delta^2 x_{t-1} &= \Delta x_{t-1} - \Delta x_{t-2} , \\ \Delta x_{t-2} &= x_{t-2} - x_{t-3} , \\ \Delta^n x_{t-1} &= \Delta^{n-1} x_{t-1} - \Delta^{n-1} x_{t-2} .\end{aligned}$$

Lower order polynomials have been of particular interest in the literature and the so-called *zero-order polynomial predictor* is being incorporated in some experimental compression system for space telemetry (Reference 13). The zero-order predictor is given by

$$\hat{x}(t) = x(t - \Delta t) .$$

In practice a tolerance may be placed around this estimate, creating a window which is equal to, or a multiple of, the quantization interval. An algorithm for the zero-order predictor would read as follows:

Zero-Order Predictor Algorithm

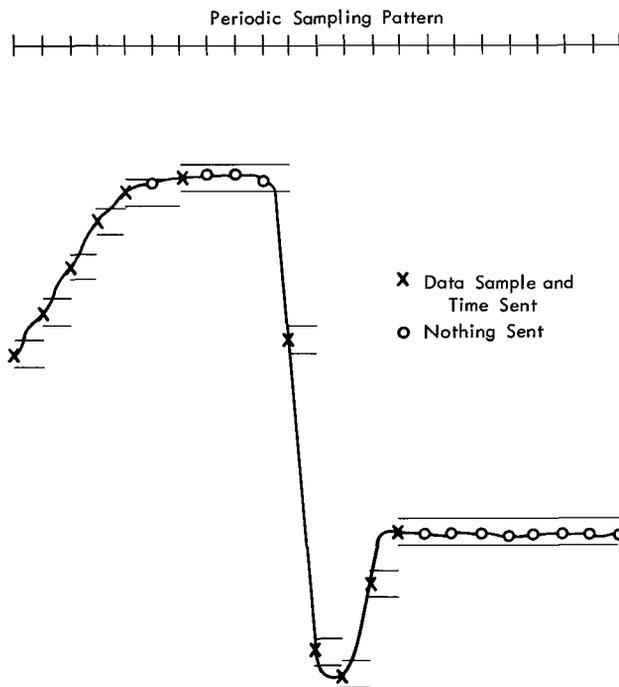


Figure 2—Zero-order predictor data compression operation.

1. Store and transmit first sample x_0 and time of occurrence.

2. Put tolerance λ about x_0 to obtain an aperture:

$$x_0 - \lambda < \hat{x} < x_0 + \lambda .$$

3. Is next sample within aperture?

If yes: discard sample and check next sample.

If no: store and transmit sample and time of occurrence and repeat steps 2 and 3.

This algorithm is illustrated in Figure 2.

The *first-order predictor* (Reference 9) is given by

$$\hat{x}_t = x_{t-1} + \Delta x_{t-1} ,$$

here

$$\Delta x_{t-1} = x_{t-1} - x_{t-2}$$

In the implementation of this first-order predictor, the actual algorithm may take on a number of different forms depending upon the definition of x_{t-1} and Δx_{t-1} in terms of tolerances. If we assume that $[x_{t-1} + \Delta x_{t-1}]$ has a tolerance λ placed about it, equal to or greater than the quantization level, then we can state the following.

First-Order Predictor Algorithm

1. Store and transmit first sample x_i and time of occurrence.
2. Store and transmit second sample x_{i+1} .
3. Compute $x_{i+1} - x_i$.
4. Add $n(x_{i+1} - x_i)$ to last transmitted sample value x_{i+1} , giving $\hat{x} = x_{i+1} + n(x_{i+1} - x_i)$, where $n = 1$ initially.
5. Place tolerance around \hat{x} so that an aperture is obtained:

$$x_{i+1} + n(x_{i+1} - x_i) - \lambda < \hat{x} < x_{i+1} + n(x_{i+1} - x_i) + \lambda$$

6. Is next sample x_{i+2} within aperture?
 - If yes: discard sample, replace n by $n + 1$, and repeat Steps 4, 5 and 6, replacing i by $i + 1$ in Step 6.
 - If no: repeat steps 1 through 6 above considering x_{i+2} the "first sample," that is replacing i by $i + 2$, and letting $n = 1$.

This algorithm is illustrated in Figure 3. In the operation of this algorithm, when a sample does not fall within the predicted zone (equal to the predicted value \pm the tolerance), a new prediction line is started using this sample and the next sample. It is also possible to start the new prediction line by using the

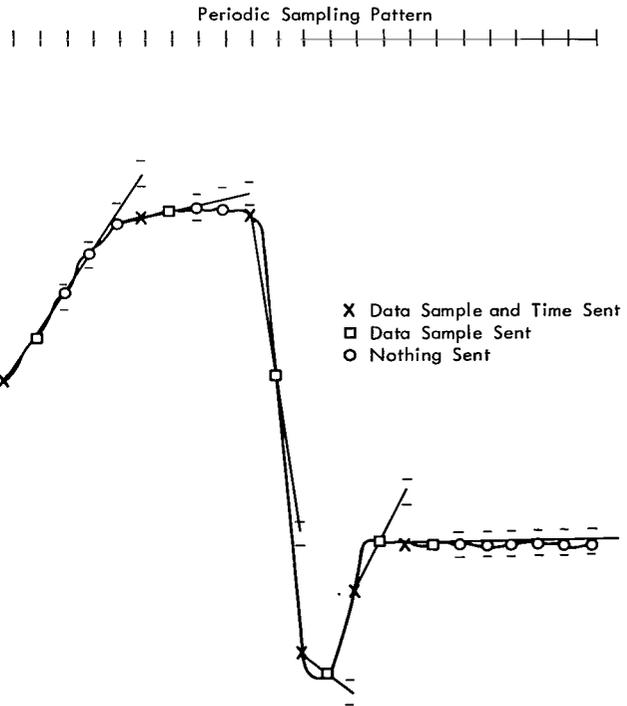


Figure 3—First-order predictor data compression operation.

last successfully predicted sample and the sample after it—in this case the sample falling outside the prediction zone. Schemes have also been proposed (Reference 8) whereby a variable tolerance is used depending upon the number of redundant samples in a row.

There is no proven optimum technique among those mentioned above, and results of computer simulations (Reference 13) of these algorithms operating on actual telemetry data do not point to a general rule for determining the optimum technique.

Coding

Coding has been divided into two main categories (Reference 10): source encoding and channel encoding.

Source Encoding

Source encoding has been aimed mostly at the efficiency of the *source-to-channel-input* part of the communication system. Efficiency has been defined as a ratio:

$$\text{Efficiency} = \frac{H_A(x)}{\bar{n}_A} .$$

\bar{n}_A is the average number of symbols per message:

$$\bar{n}_A = \sum_{i=1}^{M_d} n_i p_i \text{ for a source of } M_d \text{ messages,}$$

$$H_A(x) = \frac{H(x)}{\log_2 A} ,$$

$H(x)$ is the entropy of the source in bits,

A is the number of symbols in the encoding alphabet (in the binary case, $A = 2$).

Shannon (Reference 14) has proven the following theorem which defines the most efficient coding of a source (from the standpoint of number of transmitted symbols).

$$\frac{H(x)}{\log A} \leq \bar{n}_A < \frac{H(x)}{\log A} + 1 .$$

Some source encoding techniques which increase the efficiency defined above are *Shannon-Fano Coding* (Reference 10), *Huffman Coding* (Reference 11), *Binary Run-Length Coding* and *Sequence Coding*.

Channel Encoding

Channel encoding has been used to control the error probability in transmission over a given channel in a more efficient manner than simply increasing the average signal power corresponding to each message.

The *Noisy Channel Coding Theorem* was stated by Shannon (Reference 14) as follows:

Let a discrete channel have the capacity C and a discrete source the entropy per second H . If $H \leq C$ there exists a coding system such that the output of the source can be transmitted over the channel with an arbitrarily small frequency of errors (or an arbitrarily small equivocation). If $H > C$ it is possible to encode the source so that the equivocation is less than $H - C + \epsilon$ where ϵ is arbitrarily small. There is no method of encoding which gives an equivocation less than $H - C$.

The signalling rate then can be close to the channel capacity, and this is the significant part of the theorem: that error-free communication is possible at non-zero, reasonably fast signalling rates.

Some deterministic methods allowing error control at useful signalling rates are the Hamming Code (Reference 15), Bose-Chaudhuri Code (References 16, 17), Single Parity Code (Reference 15), Cyclic Code (Reference 18), and the Recurrent Code (Reference 18). All are block codes, as will be described below, except the Recurrent Code. The Hamming Code will be used as a typical error-control code in following chapters, and a development of its coding and decoding operations is given in Appendix A along with a plot showing its error-control performance under different bit-error probabilities for the binary-symmetric channel.

The Concept of Distance

The error-detecting and error-correcting capabilities of a code can be predicted by use of the binary *distance* property (Reference 15) which is simply a measure of the number of bit positions that differ between two code words.

The number of n -bit code words with minimum distance d (odd) has an upper bound as follows:

$$\text{No. of code words, } M_d \leq \frac{2^n}{\sum_{i=0}^{(d-1)/2} \binom{n}{i}} .$$

If we assign a certain number k of the n -bit positions to information, we then have a block code, sometimes represented by (n, k) where n is the total number of bits in the word and k of them are information bits. Then $M_d = 2^k$.

In a typical coding problem, we know the number M_d of messages to encode. We find from channel considerations, how many errors per word we wish to detect and/or correct. This establishes the minimum distance d_{min} , since (Reference 18)

$$d_{min} = e_d + 2e_c + 1 ,$$

where e_c is the number of errors to correct, and e_d is the number of additional errors that can be detected but not corrected.

Then the code word length n can be found from the M_d inequality. The task of finding ways to pick the M_d code words from the 2^n set of possible words has received considerable attention. Work is continuing, particularly in the direction of finding codes which are simple to decode.

Rate Distortion

The concept of allowable distortion and the maximum signalling rates (in the telemetry case, words per second) that we can ever theoretically obtain over a channel with capacity C (bits per second) was formulated by Shannon (Reference 12) in the *rate distortion function*.

Shannon used a distortion-measure matrix which gave the cost of making an error in the over transmission from data source set M to data user set Z . This matrix can be represented as follows for k -bit words:

$$D(M, Z) = \begin{matrix} & m_0 & m_1 & m_2 \cdots \cdots \cdots m_{2^k} \\ \begin{matrix} z_0 \\ z_1 \\ z_2 \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ z_{2^k} \end{matrix} & \begin{bmatrix} c(m_0, z_0) & c(m_1, z_0) & c(m_2, z_0) \cdots \cdots c(m_{2^k}, z_0) \\ c(m_0, z_1) & c(m_1, z_1) \cdots \cdots \cdots \cdots \cdots \cdots \\ c(m_0, z_2) \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ c(m_0, z_{2^k}) \cdots \cdots \cdots \cdots \cdots c(m_{2^k}, z_{2^k}) \end{bmatrix} \end{matrix}$$

In the usual sense of errors and cost, one could consider $c(m_i, z_i)$ as the cost of no error, and typically this would be zero. Each element off the main diagonal in the $D(M, Z)$ matrix could then be made a function of its distance from the main diagonal, or in other words the *error* in sending m_i and receiving z_j instead of z_i . No elements of $D(M, Z)$ are negative in order to have zero average distortion associated with perfect transmission.

The average distortion is then given by

$$D = \sum_{M, Z} P(M) P(Z/M) D(M, Z) ,$$

where $P(M)$ is the first-order probability of source M , and $P(Z/M)$ is the transitional probability matrix for the transmission of m_i to z_j . To find the maximum usable signalling rate over a channel of capacity C (bits/sec), Shannon assumed a bit transmission rate equal to C , and defined:

$$\text{Max. Signalling Rate} = \frac{C}{R(D)} \quad (\text{words/sec}) ,$$

where $R(D)$ is the rate distortion function in (bits/word). The rate distortion function can be thought of simply as the minimum average number of bits per word that may be used for encoding to transmit data from a given source with successive words statistically independent* in a system with a given distortion measure matrix, with an average distortion not exceeding D . Shannon called $R(D)$ the minimum source rate. More formally, we have the following:

The Rate Distortion Function

$R(D)$ is the greatest lower bound of the average mutual information $I(M; Z)$ between the statistically independent input M and the output Z of an entire communication system where the distortion D is less than some specified value. The minimization of $I(M; Z)$, subject to the constraints of a given D and

$$\sum_z P(Z/M) = 1 ,$$

may be carried out in principle by variational methods. An example is worked out in Appendix B in which $R(D)$ is obtained as a function of D for a given source and a given type of distortion measure matrix.

*Statistical independence is defined by: $P(m_j, m_i) = P(m_j)P(m_i)$.

EFFECTS OF DATA COMPRESSION ON SOURCES

In this chapter, the effects of the data compression operation on the data source will be investigated. Both Entropy Reducing (ER) and Information Preserving (IP) data compression effects will be analyzed. In the particular case of IP data compression, the effect on the time information will be introduced. This latter effect will be covered in more detail in Chapter IV.

System Arrangement of Compressors

When data compression is added to a time-multiplexed telemetry system, the arrangement of the various operations can be described by means of a diagram (Figure 4). In this figure, the entropy-reducing (ER) data compressors are tied directly to the sources they compress, but this is not the case for the information-preserving (IP) compression. Since it is expected that more than one data source will require a particular IP compression algorithm, sequential switching of a number of sources into one IP compressor is included as a basic design concept. Separate temporary storage is provided in each IP compressor for each source it handles so that predicted values may be stored for each source. Also, since buffering must take place in the IP compression process, one buffer is shown which receives the output of all IP compressors. As indicated in Figure 4, the IP compression occurs after sampling and quantizing, and the timing and the source

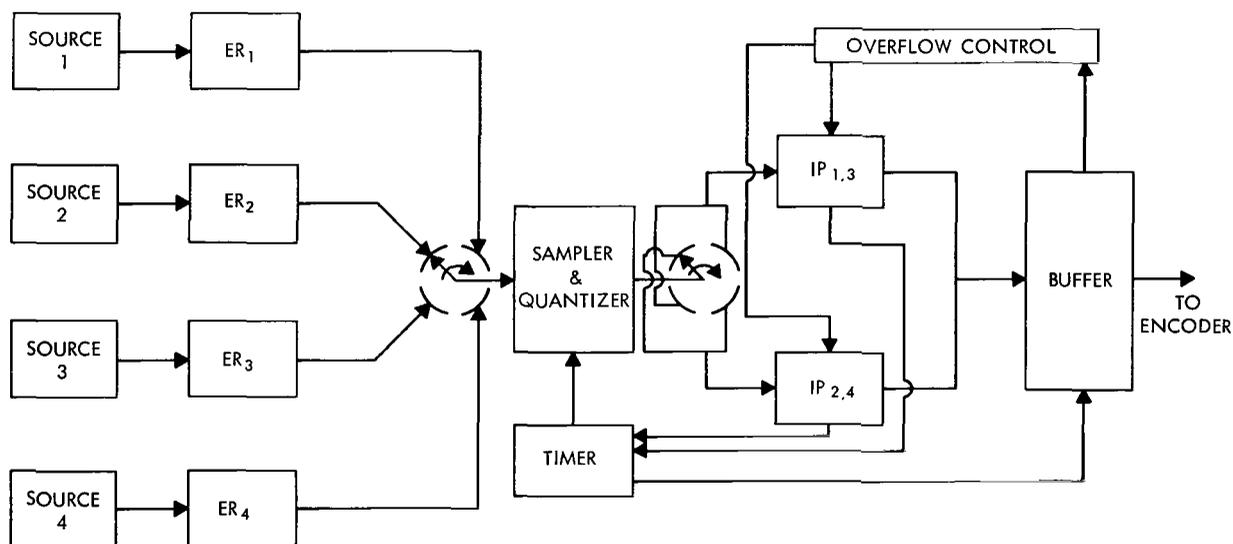


Figure 4—Conceptual diagram of a time-multiplexed telemetry system with ER and IP compression.

identification information is loaded into the buffer according to the decisions made by the individual IP compressors. Buffer overflow is prevented by feedback from the buffer to the IP compressors (References 19, 20). This feedback operation keeps track of buffer fullness and reduces the buffer input rate when overflow is imminent. A treatment of the various methods of buffer overflow control by feedback is beyond the scope of the present work.

Entropy-Reducing Compression Effects

In Chapter II, Entropy Reduction (ER) data compression is defined as an operation that reduces the fidelity of the source. In space telemetry, ER data compression normally takes the form of highly specialized data conditioning equipment on the spacecraft. There is usually a special ER device for each experiment, and sometimes it is difficult to separate the ER device and the experiment.

Effect of ER on Entropy

A relevant question at this point is whether or not ER compression always reduces the entropy. The answer can be developed as follows: If we consider the data source at the input to the ER device to have a basic limitation in *measurement* precision due to source noise and measurement hardware design, then we can express the input to the ER compressor in terms of a *discrete* source with W levels and proceed in the following way: If x is the input, and y the output of the ER device, then

$$H(X) = - \sum_{i=1}^W p(x_i) \log p(x_i) ,$$

$$H(X, Y) = H(X) + H(Y/X) = H(Y) + H(X/Y) .$$

But if $y_i = f(x_i)$, then $H(Y/X) = 0$. Also ER compression is irreversible, so $H(X/Y) > 0$. Then $H(Y) = H(X) - H(X/Y)$.

So ER compression does in fact cause a reduction in entropy, provided we consider the ER source a discrete source, finely partitioned into a finite number of levels.

Effect of ER on Source Probability Distribution

As shown in Figure 4, the ER compressor usually operates directly on the data source, before sampling and before quantization. As stated in Chapter II, the ER compressor can operate on different parameters associated with the data source—e.g., amplitude, spectrum, statistical parameters, etc. The one characteristic that all these operations have in common is entropy reduction.

Unfortunately, the knowledge that the entropy is reduced does not readily point to general effects that may be stated with regard to the probability distributions—even for those compressors

which operate on the same data parameter or characteristic. For example, in the case of the ER compressors that operate on the amplitude, it does not follow that there is a deterministic effect on the probability distribution given that the entropy is reduced, whereas by contrast there is the well-known theorem of Shannon (Reference 14) which states that any change that tends to make the probability distribution more uniform results in an increase in entropy.

For particular cases of ER compressors, the effects on the probability distributions can be worked out, and some examples are given in Appendix C for two ER compressors that operate on the data amplitude: clipping, and logarithmic amplification. As can be seen from these examples, the ER operation can have in some cases an appreciable effect on the probability distribution, and can result in highly non-uniform distributions.

Information-Preserving Compression Effects

Since Information-Preserving (IP) data compression is normally the last operation on the data before channel encoding, and since it is normally considered more a part of the telemetry system than Entropy-Reduction (ER) data compression, it is of particular interest to study the effect of this IP operation when placed into the data flow. One of the significant features of an IP operation is that it produces two kinds of output data: "non-redundant" data samples and their sample times.

In the following development the choice of an IP compressor is discussed and the question of a measure of compressibility in the case of IP compression is examined from two standpoints: redundancy removal and compressible sequences. This examination leads to a formulation of the compression ratio parameter.

Choice of a Polynomial-Predictor Type Compressor

As was noted in Chapter II, one of the classes of IP compressors which has received much attention in the development of data compression systems is the polynomial predictor type. The choice of different order polynomial predictors has been made generally (References 8, 9, 13) on the basis of an examination of waveform structure as a function of time. For data with a highly complex waveform structure, it is theoretically possible to predict long segments of the waveform using a sufficiently high-order polynomial in an adaptive mode of prediction, if necessary. However, there are two arguments against such an approach in a space telemetry system: first, there are normally limits imposed on the sophistication of the on-board data handling system by power, weight and equipment reliability considerations; second, if a waveform has a highly complicated structure of interest to some data user, it may be undesirable to compress this waveform in the first place, since it may have important small-scale properties which might be lost in the compression process. For these reasons, the simplicity as well as the efficiency of a compression algorithm are important in the space application, and the two algorithms described in Chapter II (zero- and first-order predictors) are of interest from the simplicity standpoint. This leads to the questions of what the polynomial-predictor algorithms actually do to the data from a mathematical standpoint.

The Redundancy-Removal Viewpoint

The polynomial type predictors mentioned above depend on some predictable characteristic of the data in order to achieve compression. One can find in the literature (References 5, 8) reference to Information-Preserving data compression (as used here) as Redundancy-Removing data compression. This latter terminology suggests the following viewpoint.

Shannon (Reference 14) has defined redundancy as follows: For M_d levels,

$$\text{Redundancy} = 1 - \frac{H}{\log M_d} ,$$

where H is actual source entropy. The actual source entropy is given by:

$$H = H(X) = \lim_{\alpha \rightarrow \infty} H(x_\alpha/x_0, x_1, \dots, x_{\alpha-1}) ,$$

where

$$H(x_\alpha/x_0, x_1, \dots, x_{\alpha-1}) = - \sum_{i_1}^{M_d} \sum_{i_2}^{M_d} \sum_{i_3}^{M_d} \dots \sum_{i_\alpha}^{M_d} p(x_{i_1}, x_{i_2}, \dots, x_{i_\alpha}) \log p(x_{i_\alpha}/x_{i_0}, x_{i_1}, \dots, x_{i_{\alpha-1}}) .$$

When successive samples are statistically independent,

$$H \stackrel{e}{=} H(X)_{si} = - \sum_{i=1}^{M_d} p(x_i) \log p(x_i) , \quad \text{and} \quad H(X)_{si} \geq H(X) .$$

If we think of an Information-Preserving data compression operation "removing redundancy," then the expression for redundancy must be reduced. This can happen only by an increase in the entropy, since $\log M_d$, the maximum entropy, cannot change (for a given number of levels). Mathematically: Using x as input, and y as output of the IP compressor:

Before compression

$$(R)_x = (\text{Redundancy})_x = 1 - \frac{H_x}{\log M_d} .$$

After compression

$$(R)_y = (\text{Redundancy})_y = 1 - \frac{H_y}{\log M_d} .$$

But $(R)_y < (R)_x$; thus $H_y > H_x$.

The entropy increase is actually due to the fact that the dependence between successive samples has been decreased in some instances, namely, in those instances where prediction is successful. If it were possible to make all samples statistically independent ($p(y_i, y_j) = p(y_i) p(y_j)$), then

$$\text{Minimum Redundancy } (R)_y = 1 - \frac{\left| \sum_{i=1}^{M_d} p(y_i) \log p(y_i) \right|}{\log M_d}.$$

The maximum *compression ratio* has been defined in terms of redundancy (Reference 5) as follows:

$$\text{CR} = \frac{1}{1 - (R)_x} = \frac{\log M_d}{H}.$$

If we had a source with minimum possible redundancy,

$$\text{CR} = \frac{\log M_d}{\left| \sum_{i=1}^{M_d} p(x_i) \log p(x_i) \right|},$$

which indicates that such a source could be compressed if $p(x_i)$ were not uniform. However, the predictor type of IP compressor makes successive samples more independent, and may or may not make the probability distribution $p(x_i)$ more uniform. The knowledge that IP compression increases the entropy of the data samples does not lead to a deterministic effect on the probability distribution—just as in the case of ER compression, the knowledge of an entropy *decrease* did not help in this regard either. In the case of the predictor-type algorithms, a more suitable formula for maximum compression ratio would be

$$\text{CR} = \frac{\left| \sum_{i=1}^{M_d} p(x_i) \log p(x_i) \right|}{H}.$$

The Compressible Sequence Viewpoint

The Shannon-McMillan theorem (References 21, 22) states roughly that for an ergodic* source with entropy H , there are approximately $2^{\alpha H}$ high probability α -sample sequences, each with a probability of about $2^{-\alpha H}$, and that the remaining group of sequences, which includes the vast majority of possible α -sample sequences, has a low total probability. Mathematically (Reference 22),

*The ergodic property is taken here to mean the following: the frequency of occurrence of a given α -sample sequence in a long temporal span of data converges to the probability of the sequence (Reference 21).

given arbitrarily small $\epsilon > 0$ and $\delta > 0$, it is possible to find an α such that any sequence S among the $2^{\alpha H}$ highest probability sequences has a probability P_s for which

$$\left| \frac{\log P_s}{\alpha} + H \right| < \epsilon ,$$

and such that the total probability of all other sequences is less than δ .

If we consider the data to be compressed an ergodic source, we can look at α -sample sequences which include those sequences that are compressible by the algorithm in question. From the stand-point of compression, we would like α to be large, and also the total number N_c of compressible sequences to represent a large number of the high probability sequences. The ideal compressible source will then have the following breakdown of α -sample sequence types:

$$\text{Total number of sequences} = (M_d)^\alpha .$$

$$\text{Total number of high probability sequences} \approx 2^{\alpha H} .$$

$$\text{Total number of compressible sequences} = N_c < 2^{\alpha H} .$$

Then a formula for the maximum compression ratio could be written as

$$\text{CR} \approx \frac{2^{\alpha H}}{2^{\alpha H} - N_c} = \frac{1}{1 - \frac{N_c}{2^{\alpha H}}} .$$

This formula resembles the one given above, viz.

$$\text{CR} = \frac{1}{1 - (R)} ,$$

and now redundancy can be expressed as

$$(R) = \frac{N_c}{2^{\alpha H}} .$$

This is a more suitable formula for redundancy in the predictor type of compression operation for two reasons: it does not involve the entropy of a uniform probability source, and it is given as the percentage of the *high-probability* sequences which are compressible by the particular algorithm under consideration.

Actual Compression Ratio

In an actual system, the above ideal expressions for compression ratio serve only as upper bounds since they do not take into consideration the timing data that must be sent or the extra bits required for error-control coding.

A useful term for actual compression ratio is "bit-compression ratio," defined as

$$\text{CR} = \frac{\text{No. of bits to send (uncompressed data)}}{\text{No. of bits to send (same data compressed)}} .$$

This actual compression ratio takes into account *all* bits required—including data, time, identification and coding.

Chapter IV

THE TIME ENCODING PROBLEM

In this chapter an essential part of the IP data compression system will be examined—time encoding. Modified versions of the polynomial predictor algorithms described in Chapter II which require fewer time words are introduced. Three methods of time encoding will be analyzed and compared on a noise-free basis. For one method, sequence coding, an encoding and decoding algorithm will be developed. The effect of noise on the time information will be covered in Chapter V.

Nature of the Problem

An IP data compression operation normally requires that the time of each non-redundant sample be transmitted with the sample value. In an uncompressed system the time information is sent periodically, and for the sake of description it may be represented as a monotonically increasing function of transmission time with periodic, equal discontinuities. In a data compressed system, timing information does not have equal discontinuities, but is still monotonically increasing.

It is significant to note that for the predictors shown in Chapter II, the time information need not be sent with each non-redundant sample. Since non-redundant sample values and time values are loaded into a buffer they may be read out in any convenient fashion. For example, all sample values could be read out and then the time words could follow as a block. In subsequent processing, corresponding samples and times could be matched by a simple counting process. Just as the sample values and time words can be read out separately, so also can they be encoded and decoded separately.

If the data-compressed telemetry system were handling one source, then data values and time values would be required for transmission. A system with more than one data source would require data values, time values and source identification words for transmission. However, if the sampling of the various sources were time-shared as indicated in Figure 4, then it would be no longer necessary to send both source identification and non-redundant sample times. These two pieces of information could be provided by merely sending the *position* of each non-redundant sample in the time-shared sampling pattern with respect to a periodic reference position in the pattern. This time-shared system is used in the analysis of various time encoding schemes in this chapter, and some of the same terms are used here as were used for the time-multiplied telemetry pattern described in Chapter II. This consistency of terminology is for convenience—it does not mean we are compressing a sequence of multiplexed samples, but that we are compressing each source in time according to a time-shared (time-multiplexed) pattern by an arrangement such

as the one shown in Figure 4. In this sense, we use the term "compressed time-multiplexed data."

It should be pointed out that the time-shared sampling with fixed-length data words and fixed-length time words used here has the advantages of simplicity and compatibility with present-day space telemetry systems, but it is not the only sampling arrangement that is amenable to data compression. For example, a system with variable sampling rates for individual sources or groups of sources could be used, and this would typically require additional buffering, and variable-length identification and time-prefixes to the data words. The concepts and techniques of source encoding mentioned in Chapter II could be applied to both the identification, and time-prefixes in such a system, thereby realizing some additional data compression.

Modified Predictor Algorithms

It is possible to modify* the polynomial-predictor algorithms described in Chapter II so that fewer time words will be required for data reconstruction for certain types of data behavior. In no case will these modified algorithms require more time words than those described in Chapter II. The following are the modified algorithms corresponding to the zero- and first-order predictors of Chapter II. The operations of these algorithms, particularly on the time words, can be compared to the corresponding ones in Chapter II.

Modified Zero-Order Predictor Algorithm

1. Store and transmit first sample x_0 and time of occurrence.
2. Put tolerance about x_0 such that an aperture is obtained:

$$x_0 - \lambda < \hat{x} < x_0 + \lambda$$

3. Is next sample within aperture?

If yes: discard sample and check next sample.

If no: was last sample transmitted?

If yes: transmit only sample value.

If no: transmit sample value and time of occurrence.

This algorithm is illustrated in Figure 5.

Modified First-Order Predictor Algorithm

1. Store and transmit first sample x_i and time of occurrence.
2. Store and transmit second sample x_{i+1} .

*The author is indebted to Prof. Alan B. Marcovitz for suggesting this modification.

3. Compute $x_{i+1} - x_i$.
4. Add $n(x_{i+1} - x_i)$ to last transmitted sample value, x_{i+1} giving $\hat{x} = x_{i+1} + n(x_{i+1} - x_i)$. Let $n = 1$ initially.
5. Place tolerance around \hat{x} so that an aperture is obtained:

$$x_{i+1} + n(x_{i+1} - x_i) - \lambda < \hat{x} < x_{i+1} + n(x_{i+1} - x_i) + \lambda$$

6. Is next sample x_{i+2} within aperture?
 - If yes: discard sample, replace n by $n + 1$, repeat steps 4, 5, and 6, replacing i by $i + 1$ in step 6.
 - If no: was last sample transmitted?
 - If no: repeat steps 1 through 6 above considering x_{i+2} the "first sample," that is, replacing i by $i + 2$ and letting $n = 1$.
 - If yes: follow same procedure as for *last sample not transmitted* except do not send time of occurrence in step 1.

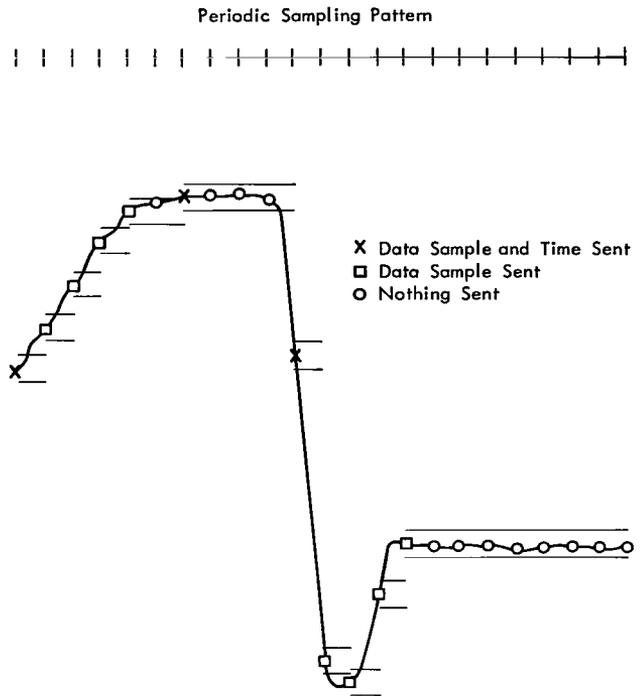


Figure 5—Modified zero-order predictor data compression operation.

As can be seen from an examination of Figure 5 the time of the *beginning* of a run of non-redundant samples is sent along with these non-redundant samples under the modified algorithms. To reconstruct the data, one merely fills in redundant (as defined by the order of the algorithm) samples after the last non-redundant sample in a run up to the beginning of the next run. However, each time word must be identified with the sample value it applies to.

In the operation of the modified algorithms, then, a flag is required to differentiate a time word from a data word, and in the simplest arrangement this could increase the time word length as well as the data word length by one binary digit.

The modified algorithms show the greatest improvement in time word economy over their unmodified counterparts when there are long runs of non-redundant data samples followed in each case by a long run of redundant data samples. The number of time words required by a modified algorithm equals that of the corresponding unmodified algorithm when, for example in the case of the zero-order predictor, the non-redundant sample runs are just one sample each, so that the redundant data sample runs are separated by single non-redundant samples. The modified algorithm technique can also be applied to a group of time-shared-sampled sources with a known sampling pattern, and this will be illustrated in the following section.

Three Basic Methods

As indicated previously, the source identification and timing information is essential for the reconstruction of compressed time-multiplexed data. There are various methods and techniques of sending and using this information, but most of them can be grouped into the following three categories:

- Identification of what is being sent.
- Identification of what is *not* being sent.
- Sequence identification.

These three methods are discussed below.

Identification of Transmitted Words

In this method each transmitted word or group of transmitted words (for the modified algorithm) is accompanied by an identifier which gives the position of the word in the telemetry minor frame. An additional word must be sent to identify the minor frame and this word is the sub-commutator count (S.C.C.). The main frame count word is also sent to keep track of the main frames. In addition, word synchronization has to be provided for, so a sync word is sent once each minor frame. In order to prevent long periods over many main frames, wherein no samples from some super-commutated sources would appear, a sample from each source is sent at least once per main frame. If there are a words per minor frame, b minor frames per main frame, and s sources (other than sync and subcom count), then

$$\begin{aligned} \text{Number of words in main frame} &= ab, \\ \text{Minimum number of words} &= \left\{ \begin{array}{l} b \text{ sync words} \\ + \\ b \text{ subcom count words} \\ + \\ s \text{ words (includes} \\ \text{main frame count)} \end{array} \right\} = 2b + s. \end{aligned}$$

Since for some sampling patterns it may not be possible to group these required words together, thereby saving on time words, it will be assumed below that a time word accompanies each one.

$$\text{Maximum overall compression ratio} = \frac{ab}{2b + s}.$$

If we use the term "word compression ratio" $(CR)_m$ to mean the ratio of the maximum number of compressible samples in a main frame to the number q_m of these samples left in the main frame after compression, then, assuming $q_m > 0$:

$$(CR)_m = \frac{ab - (2b + s)}{q_m}.$$

We can average this over many main frames and obtain $\overline{(\text{CR})}_m$. Therefore, an expression for the actual average compression ratio* $\overline{(\text{CR})}_{ma}$ on a main frame basis for the unmodified algorithms (each transmitted word is accompanied by a position word) can be written as

$$\overline{(\text{CR})}_{ma} = \frac{ab(k)}{\left\{ (2b + s) + ab \frac{(2b + s)}{\overline{(\text{CR})}_m} \right\} (k + k_t)}$$

where k is the number of bits/word for data, and k_t is the number of bits/word for position in minor frame.

In many telemetry systems $a = b$, so that we may write for this case:

$$\overline{(\text{CR})}_{ma} = \frac{a^2 (k)}{\left\{ 2a + s + a^2 \frac{(2a + s)}{\overline{(\text{CR})}_m} \right\} (k + k_t)}$$

Now the number of bits required for position information per transmitted word is related to the minor frame length a as follows:

$$k_t = \log_2 a \quad (\text{increased to nearest integer}).$$

Then

$$\overline{(\text{CR})}_{ma} = \frac{a^2}{\left(2a + s + a^2 \frac{(2a + s)}{\overline{(\text{CR})}_m} \right) \left(1 + \frac{\log_2 a}{k} \right)}$$

In a given telemetry system, k is usually fixed, so that the above expression relates an actual average compression ratio $\overline{(\text{CR})}_{ma}$ to a theoretical average compression ratio $\overline{(\text{CR})}_m$ for various values of main frame size a and number s of compressible sources.

For the modified algorithm case where it is not necessary to identify the position of every transmitted word, but only the position of the beginning of a contiguous sequence of transmitted words in the pattern by a time word which contains an extra bit for identification, the above formula is modified by a factor K on $(k_t + 1)$ such that $0 < K \leq 1$. Also, letting $k - k + 1$,

$$\overline{(\text{CR})}_{ma} = \frac{a^2}{\left(2a + s + a^2 \frac{(2a + s)}{\overline{(\text{CR})}_m} \right) \left(1 + \frac{1}{k} + \frac{K(1 + \log_2 a)}{k} \right)}$$

*As described in Chapter II, actual average compression ratio is the ratio of the number of binary digits required for the uncompressed system to the number of binary digits required for the compressed system.

Now K has a maximum value equal to 1 when single transmitted samples alternate with runs of non-transmitted samples. Then each transmitted sample must be accompanied by a position indicator. The value of K never reaches zero since even if all samples from each source were redundant, the position of the *first* sample from each source in the main frame would have to be sent (s positions) as well as the position of the synchronization word and sub-commutator count words (a positions—assuming one follows the other). Then the minimum value of K would be the fractional part of all transmitted words that corresponds to $a + s$, or

$$K_{\min} = \frac{a + s}{\left(2a + s + \frac{a^2 - 2a - s}{(\overline{CR})_m}\right)} .$$

At this point, it is interesting to examine the factor K as a function of $(\overline{CR})_m$. In order to get a feel for this function, we assume a source with statistically-independent successive samples. Now although this source does not represent the compressible type as explained in Chapter III it gives rise to a simple model for the K variation with $(\overline{CR})_m$.*

For this source, assume two kinds of samples: non-redundant and redundant. Then, on an average basis:

$$\text{Probability of a non-redundant sample} = \frac{1}{(\overline{CR})_m} .$$

$$\text{Probability of a redundant sample} = 1 - \frac{1}{(\overline{CR})_m} .$$

Now it is necessary to send a time word in the modified algorithm whenever we proceed from a redundant data word to a non-redundant data word (see Figure 5).

The probability of a time word transmission is given by:

$$\text{Prob. (time word)} = \text{Prob. (non-redundant/redundant)} \times \text{Prob. (redundant)} ,$$

or

$$\text{Prob. (time word)} = \frac{1}{(\overline{CR})_m} \left[1 - \frac{1}{(\overline{CR})_m} \right] .$$

Now K can be defined as

$$K = \frac{\text{number of transmitted data samples that need time words}}{\text{number of transmitted data samples}} ,$$

*The author is indebted to Prof. Alan B. Marcovitz for suggesting this model and associated analysis.

or another way,

$$K = \frac{\text{probability of a non-redundant sample that needs a time word}}{\text{probability of a non-redundant sample}},$$

$$K = \frac{\frac{1}{(\overline{\text{CR}})_m} \left[1 - \frac{1}{(\overline{\text{CR}})_m} \right]}{\frac{1}{(\overline{\text{CR}})_m}} = 1 - \frac{1}{(\overline{\text{CR}})_m}.$$

Identification of Non-Transmitted Words

In this system, we send along with each transmitted word or group of words (for the modified algorithm), the count of the number of multiplexed words not transmitted since the previous transmitted word.

If we add a word to each transmitted word which signifies the total number of *multiplexed* words *not* transmitted since the last transmitted word, we would have a time word of length $\lceil \log_2 (a - 2) \rceil$ bits, since at least two words are transmitted per minor frame (sync and S.C.C.) and if one followed the other, then the maximum number of non-transmitted words would be obtained, equal to $a - 2$ (assuming there is at least one minor frame with no data samples).*

We can then write formulas for actual average compression ratio similar to those above. For the unmodified algorithms,

$$(\overline{\text{CR}})_{ma} = \frac{a^2}{\left(2a + s + \frac{a^2 - 2a - s}{(\overline{\text{CR}})_m} \right) \left(1 + \frac{\log_2 (a - 2)}{k} \right)}.$$

For the modified algorithms,

$$(\overline{\text{CR}})_{ma} = \frac{a^2}{\left(2a + s + \frac{a^2 - 2a - s}{(\overline{\text{CR}})_m} \right) \left(1 + \frac{1}{k} + K \frac{\lceil 1 + \log_2 (a - 2) \rceil}{k} \right)}.$$

Sequence Encoding of Timing Information

The timing information of a single data-compressed source or a time-multiplexed group of data-compressed sources exhibits an interesting property, namely, *strict-monotonicity*. This property suggests the encoding of sequence of *time* words or multiplexed minor frame *position* words into unique code words.

*It is also possible to shorten the time word to $\lceil \log_2 (a - 2) - 1 \rceil$ if the system is arranged to send a data sample at least once every half minor frame.

The following description of sequence identification is purposely left general so that it can be applied either to the output of a *single* source with strictly-increasing monotonic time, or to the output of a *time-multiplexed* group of sources with strictly-increasing monotonic minor frame position. In order to obtain a measure on the efficiency of this sequence coding we examine a monotonic source in the following way.

For a monotonic source model to describe the timing information, we assume a total of m intervals between the periodic timing words, which timing words are transmitted whether or not compression is taking place. This leaves a maximum of $m - 1$ data words between periodic time words.

When compression takes place, q of the $m - 1$ words are left to be transmitted and the problem remains of indicating *which* q words of the $m - 1$ words are actually being sent. In the single source case, the *time* of each of the q words is the parameter of interest. In the multiplexed source case, the *identification and time* of each of the q words are of interest. But in a fixed format time-multiplexed system, the timing information can be obtained from the position of the word in the minor frame, and therefore the problem is essentially the same in both cases. The sequence to be identified is a strictly-increasing monotonic sequence of numbers from 1 to $m - 1$.

Theoretically, there would be a maximum number

$$\sum_{q=0}^{m-1} \binom{m-1}{q} = 2^{(m-1)}$$

of sequences, and this can be shown as follows.

The compression operation will result in the omission of zero or more sample times from the interval of m sample periods excluding the beginning and end times which are always transmitted. Then we can state that the number N_T of possible time sequences is equal to the total number of combinations of $m - 1$ samples taken 0, 1, 2, . . . , $(m - 1)$ at a time, or

$$N_T = \sum_{q=0}^{m-1} \binom{m-1}{q} .$$

The above expression for N_T is also the expression for the sum of the coefficients in a binomial expansion of the form $(x + y)^n$.

If we let $x = 1$ and $y = 1$, then

$$2^n = \sum_{q=0}^n \binom{n}{q} .$$

In the present case, $n = m - 1$. Substituting,

$$N_T = \sum_{q=0}^{q=m-1} \binom{m-1}{q} = 2^{(m-1)}.$$

In an actual data compression system, q is known, the number of data words left between periodic time words. Then

$$\left\{ \begin{array}{l} \text{Total number of strictly-} \\ \text{increasing sequences } q\text{-words} \\ \text{long} \end{array} \right\} = N_q = \binom{m-1}{q},$$

since the decision operation of compression is to send or not to send a particular source word.

It is obvious that

$$\text{Word Compression Ratio} = (CR)_w = \frac{m-1}{q}.$$

The number N_q of sequences is plotted (Reference 23) vs. the word compression ratio $(CR)_w$ in Figure 6 for different values of $m - 1$ as well as the number of bits to encode N_q , given by $\log_2 N_q$. The number $q \log_2 (m - 1)$ of bits to encode all q time words is also plotted.

The matter of actually encoding and decoding these sequences can be examined by slightly modifying a method developed by Gordon (Reference 24) for non-decreasing monotonic sources. In Gordon's method, a *path-count matrix* is constructed from which a *coding matrix* is derived, which is used for both encoding and decoding. The same general procedure is used here. However, the resulting matrices are different from those of the non-decreasing monotonic source.

The various q -length sequences out of $m - 1$ elements may be represented as a matrix of points as shown in Figure 7a. This figure is drawn for an actual case of $m = 9$ and $q = 6$. This is obviously not a good example of worthwhile compression, but these numbers are used for convenience in illustrating the techniques involved.

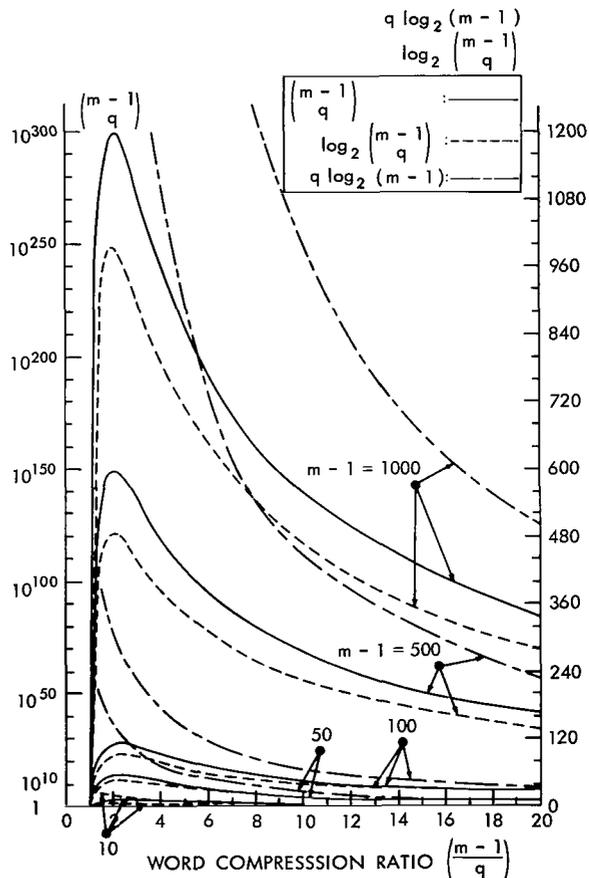


Figure 6—Plots of $\binom{m-1}{q}$, $\log_2 \binom{m-1}{q}$, and $q \log_2 (m - 1)$ vs. $\frac{m-1}{q}$ for various values of $m - 1$.

3. All elements below this diagonal are zero.

4. If the rows are numbered in an increasing direction from the bottom to the top, and the columns from left to right, then the value of the element $a_{i,j}$ is computed by the formula:

$$a_{i,j} = a_{i-1,j} + a_{i-1,j-1}$$

It should also be noted that the coding matrix indicated in Figure 7b would again be limited to the boundaries imposed by the two diagonals from the *start* and *end* corners. In this case ($m = 9$, $q = 6$) these are the all-zero diagonal below the unity diagonal, and the diagonal just above the unity diagonal with elements 2, 3, 4, 5, 6, 7.

To encode a sequence, draw the path on the coding matrix and add the elements touched by the path. This sum provides a unique code word w whose value ranges from 0 to $N_q - 1$ inclusive.

To decode a received code word w , start at the element $a_{i,q}$ in the rightmost column of the coding matrix which satisfies the inequality

$$a_{i,q} \leq w < a_{i+1,q}$$

Now $a_{i,q}$ is on the sequence path. To find the next point on the sequence path, use $w_{-1} = w - a_{i,q}$ and find $a_{i,q-1}$ such that

$$a_{i,q-1} \leq w_{-1} < a_{i+1,q-1}$$

Now $a_{i,q-1}$ is on the sequence path. Continue this procedure until a point on the path in the first column has been found. Then the sequence is given by the set of values corresponding to the row numbers of the elements in the q columns, starting at the leftmost column.

This encoding and decoding procedure is illustrated in Figure 8 for the case of $m = 9$, $q = 6$, and a particular sequence.

Comparison of the Three Methods

In order to compare the three methods discussed above, we can plot curves of *Actual Average Compression Ratio vs. Theoretical*

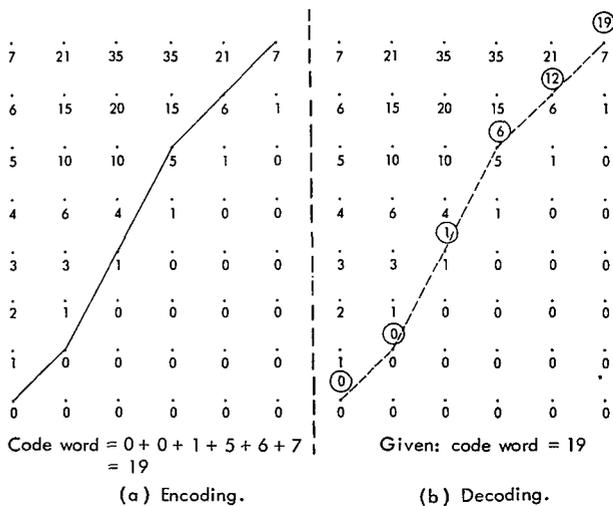


Figure 8—An example of sequence encoding and decoding.

Average Word Compression Ratio. Consistent formulas (for these parameters) were developed for the first two of the three methods, and now it remains to put the third method, sequence identification, into a form consistent with the other two.

We define an average word compression ratio for the sequence case as

$$\overline{(\text{CR})}_w = \frac{\bar{c}}{\bar{q}},$$

where \bar{q} is the average number of compressible words transmitted per minor frame, and \bar{c} is the average number of compressible words per minor frame. Then

$$\bar{c} = a - 2 - \frac{s}{b},$$

and

$$\overline{(\text{CR})}_w = \frac{\left[a - 2 - \frac{s}{b} \right]}{\bar{q}} \left(\frac{b}{b} \right),$$

or

$$\overline{(\text{CR})}_w = \frac{ab - 2b - s}{q_m} = \overline{(\text{CR})}_m.$$

Thus we can use $\overline{(\text{CR})}_w$ or $\overline{(\text{CR})}_m$ interchangeably. For the sequence case, for a square pattern, and a sequence word every minor frame,

$$\overline{(\text{CR})}_{ma} = \frac{a^2}{\left(2a + s + \frac{a^2 - 2a - s}{\overline{(\text{CR})}_m} \right) + \frac{a}{k} \log_2 \left(\frac{a - 2}{\overline{(\text{CR})}_m} \right)}.$$

Now we may summarize the formulas for plotting as follows.

a. For Identification of Transmitted Words Method

$$\overline{(\text{CR})}_{ma} \text{ (unmodified)} = \frac{a^2}{\left(2a + s + \frac{a^2 - 2a - s}{\overline{(\text{CR})}_m} \right) \left(1 + \frac{\log_2 a}{k} \right)}.$$

$$\overline{(\text{CR})}_{ma} \text{ (modified)} = \frac{a^2}{\left(2a + s + \frac{a^2 - 2a - s}{\overline{(\text{CR})}_m} \right) \left(1 + \frac{1}{k} + K \frac{(1 + \log_2 a)}{k} \right)}.$$

b. For Identification of Non-transmitted Words Method

$$\overline{(\text{CR})}_{ma} \text{ (unmodified)} = \frac{a^2}{\left(2a + s + \frac{a^2 - 2a - s}{\overline{(\text{CR})}_m}\right) \left(1 + \frac{[\log_2 (a - 2)]}{k}\right)}$$

$$\overline{(\text{CR})}_{ma} \text{ (modified)} = \frac{a^2}{\left(2a + s + \frac{a^2 - 2a - s}{\overline{(\text{CR})}_m}\right) \left(1 + \frac{1}{k} + K \frac{[1 + \log_2 (a - 2)]}{k}\right)}$$

c. For Sequence Encoding or Identification Method

$$\overline{(\text{CR})}_{ma} = \frac{a^2}{\left(2a + s + \frac{a^2 - 2a - s}{\overline{(\text{CR})}_m}\right) + \frac{a}{k} \log_2 \left(\frac{a - 2}{\overline{(\text{CR})}_m}\right)}$$

Comparison curves are plotted in Figure 9 with K equal to three values: maximum (unity), minimum, and $[1 - 1/\overline{(\text{CR})}_m]$.

From the curves in Figure 9, the following observations may be made: (Efficiency is used here as $\overline{(\text{CR})}_{ma}/\overline{(\text{CR})}_m$.)

Due to the need for always sending particular words in the pattern, a larger pattern or main frame will be more efficient at useful word compression ratios (5 or more).

Sequence coding is always more efficient than the unmodified algorithms sending either non-redundant times (method a) or redundant times (method b).

As the main frame size increases from 10×10 to 100×100 , the non-redundant-time method (a) becomes indistinguishable from the redundant-time method (b).

At small frame size (10×10), sequence coding is most efficient.

As the frame size is increased, the modified algorithms typically become as efficient as sequence coding. In the particular case where non-redundant samples are grouped together in a sequence of samples ($k = \text{min. value}$), the modified algorithms are more efficient than sequence coding.

Two important factors related to sequence coding that are not evident from the curves of Figure 9 are the following:

Sequence coding requires more complex equipment to encode the words (typically 100 bits) involved.

Errors in a sequence code will have a more catastrophic effect on the reconstructed data than errors in methods a or b.

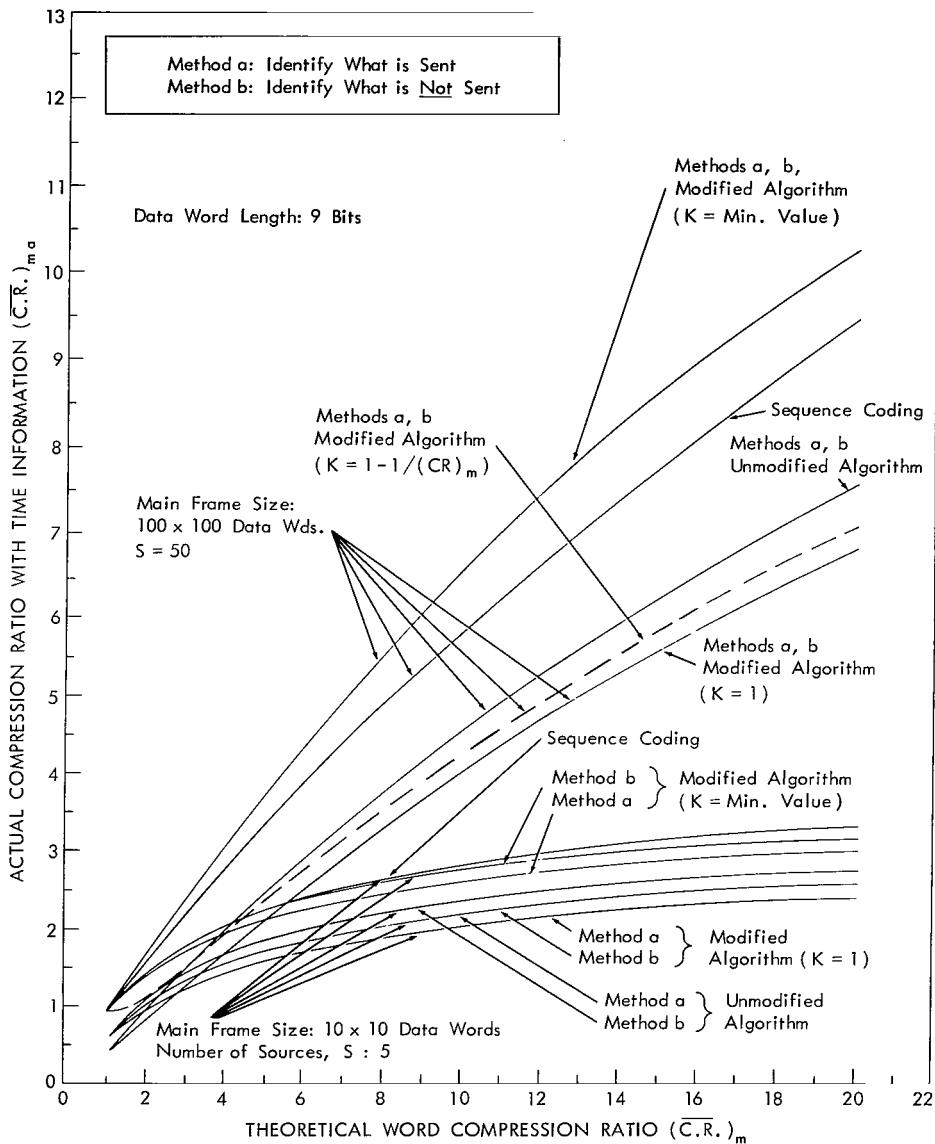


Figure 9—Comparison of three time encoding methods for time multiplexed compressed telemetry patterns.

In view of these serious limitations of sequence coding, it may be an undesirable time encoding method in many systems, despite its relatively high efficiency.

The modified algorithms do not allow the flexibility of encoding provided by the unmodified algorithms since the time words cannot be sent separately in a block. However, the typically higher efficiency of the modified algorithms over the unmodified algorithms make them well worth considering in a particular system design.

The effect of errors in the time information on the system performance will be covered in Chapter V.

Chapter V

TRADEOFF MEASURES

In the design of a space telemetry system, the decision to use data compression has to be justified by prediction of an overall gain in system performance. In situations where data may have little redundancy, or where channel disturbances may require extensive error-control coding of compressed data, it may not be feasible to include data compression in the overall design. Since data compression of the Information-Preserving type may require error-control coding, it is important to make tradeoff analyses among various compressed and coded systems as well as the uncompressed and uncoded system. Measures of performance which are consistent with the effects of data compression and error-control coding on a space telemetry system are needed in order to carry out these comparisons.

In this chapter, two measures of performance are developed for the comparison of three system types: uncompressed, uncoded; compressed, uncoded; and compressed, coded. The effect of errors in the time information is also considered and worked into the performance measure.

Measures Based on the Rate Distortion Concept

The *rate distortion function* of Shannon (Reference 12) is the basis for the measures of performance developed in this Chapter. In Chapter II, it was pointed out that the rate distortion function is the minimum mutual information between the input and the output of an *entire* communication system subject to the constraint that the average distortion be less than some specified value. For a given distortion measure matrix $D(M, Z)$, a given source probability distribution $P(M)$ for statistically independent words, and a given maximum value of average distortion D , the minimization of the mutual information can lead to a specification of the $P(Z/M)$ matrix over the entire system—from data *source* set M to data *user* set Z . However, this minimization can be carried out explicitly only for certain types of distortion measures and source probabilities (Reference 12)—such as the example worked out in Appendix B. Even in cases where the $P(Z/M)$ matrix may be explicitly found for the rate distortion function, the channel characteristics determined by such a $P(Z/M)$ matrix may not resemble those of any known channel model, such as the binary-symmetric channel. Channels with $P(Z/M)$ matrices other than the one corresponding to the rate distortion function will produce a source rate (bits per word) larger than the rate distortion value. For this reason then, the rate distortion function can be considered a lower bound for the average bits per word given a value of average distortion. This bound also applies to IP compression where we approach statistically independent successive words.*

*Successive words could never be statistically independent for a zero-order predictor since on the output of this predictor $P(y_i, y_i) = 0$, but $P(y_i) \neq 0$ in general.

The *average distortion* D used in the rate distortion function involves parameters pertaining to the source $P(M)$, the channel $P(Z/M)$, and the eventual importance of each data point $D(M, Z)$ in the following way:

$$D = \sum_{M, Z} P(M) P(Z/M) D(M, Z) .$$

These properties make the average distortion a useful measure of performance in comparing telemetry systems with various combinations of compression and coding. Likewise, a companion measure, the *rate ratio* R in bits per uncompressed bits is useful in these system comparisons. This measure is the reciprocal of the overall compression ratio and corresponds to the source rate.

Rate-Ratios and Distortions for Three Systems

The concept of source rate and distortion as used by Shannon in his rate distortion function can be used to obtain measures of performance of systems with different combinations of compression and coding. Three systems will be considered: uncompressed, uncoded (UU); compressed, uncoded (CU); and compressed, coded (CC).

The source rate was used by Shannon as the number of bits per transmitted word. For a fixed information rate, this definition is consistent since larger values of source rate correspond to higher *bit rates*. For comparison purposes, the UU rate ratio, R_{uu} will be normalized to unity. Now the three rate-ratios can be written down in terms of bits per uncompressed bits as:

$$R_{uu} = 1 ,$$

$$R_{cu} = \frac{1}{(\overline{CR})_{ma}} ,$$

$$R_{cc} = \frac{1}{(\overline{CR})_{ma}} (K_c) ,$$

where $(\overline{CR})_{ma}$ is the actual average main frame compression ratio as developed in Chapter IV, and K_c is a factor greater than unity which accounts for data expansion due to error-control coding. From the above definitions, then

$$R_{cu} < R_{uu} ,$$

$$R_{cu} < R_{cc} .$$

Now the relationship between R_{uu} and R_{cc} is a critical part of the tradeoff analysis. If $R_{cc} > R_{uu}$, we have a system with overall data expansion rather than compression. For each case of $R_{cc} < R_{uu}$, the

question must be asked: "Are the final reductions in rate-ratio and distortion worth the compression and coding required to achieve them?"

The distortion that will be used in the tradeoff analysis here is the average distortion used by Shannon:

$$D = \sum_{M, Z} P(M) P(Z/M) D(M, Z) ,$$

where $P(M)$ is the first-order probability of source M , $P(Z/M)$ is the system transitional probability matrix for the transmission of word m_i to word z_j , and $D(M, Z)$ is the distortion measure matrix as defined in Chapter II. For consistency in comparison, the same $D(M, Z)$ matrix will be used for all three systems. In order to distinguish, in the following development, between the effects of data and time errors on D , the effect of time errors will be omitted in the beginning and introduced in the next section. The development of the average distortions D_{uu} , D_{cu} and D_{cc} proceeds as follows.

The Uncompressed-Uncoded System

The expression for D_{uu} is given by

$$D_{uu} = \sum_{M, Z} P(M)_{uu} P(Z/M)_{uu} D(M, Z) .$$

The first-order probability $P(M)_{uu}$ is that of the sampled and quantized uncompressed source. The transitional probability $P(Z/M)_{uu}$ is the conditional probability of the *data user* obtaining sample level z_j given that sample level m_i was sent. The reason the data user is specified here is the fact that only he can utilize a priori knowledge about, along with inherent redundancy in, the uncompressed data for the purpose of error control. This is an important consideration in tradeoff comparisons between an uncompressed, uncoded system and a compressed, coded system. In order to quantitatively include this error-control ability on the part of the data user, we arrange the $P(Z/M)_{uu}$ matrix to take account of this ability. One such arrangement is as follows. If the data user can detect large errors in the data, then the elements in $P(Z/M)_{uu}$ corresponding to these errors can be set equal to zero. Now, the other elements of $P(Z/M)_{uu}$ have to be suitably modified so as to insure

$$\sum_z P(Z/M)_{uu} = 1 .$$

Since $P(Z/M)_{uu}$ is a square matrix of order 2^k , the above modification corresponds to setting all elements in the upper-right and lower-left regions equal to zero.

If we think of starting with a $P(Z/M)$ matrix whose elements are simply the word probabilities over a binary symmetric channel, then we can calculate all elements of this $P(Z/M)$ and obtain $P(Z/M)_{uu}$ by setting the above corner regions equal to zero and then adding to the remaining elements in each column the total probability removed from that column. If we assume that the data user performs error correction as well as error detection, then we add the total probability removed from each column to the main diagonal element in that column. Now the relative ability of a data user to perform error detection or correction can be represented by the *size* of the corners of $P(Z/M)_{uu}$ that are set to zero. For example, for a 3-bit, 8-level system, the $P(Z/M)$ matrix corresponding to transmission over a binary-symmetric channel has the form:

$$[P(Z/M)] = \begin{array}{c|cccccccc} Z \backslash M & 000 & 001 & 010 & 011 & 100 & 101 & 110 & 111 \\ \hline 000 & q^3 & q^2 p & q^2 p & qp^2 & q^2 p & qp^2 & qp^2 & p^3 \\ 001 & q^2 p & q^3 & qp^2 & q^2 p & qp^2 & q^2 p & p^3 & qp^2 \\ 010 & q^2 p & qp^2 & q^3 & q^2 p & qp^2 & p^3 & q^2 p & qp^2 \\ 011 & qp^2 & q^2 p & q^2 p & q^3 & p^3 & qp^2 & qp^2 & q^2 p \\ 100 & q^2 p & qp^2 & qp^2 & p^3 & q^3 & q^2 p & q^2 p & qp^2 \\ 101 & qp^2 & q^2 p & p^3 & qp^2 & q^2 p & q^3 & qp^2 & q^2 p \\ 110 & qp^2 & p^3 & q^2 p & qp^2 & q^2 p & qp^2 & q^3 & q^2 p \\ 111 & p^3 & qp^2 & qp^2 & q^2 p & qp^2 & q^2 p & q^2 p & q^3 \end{array} ,$$

$$q = 1 - p ,$$

where p is the probability of a bit error. For a data user who can *correct* errors greater than 4 levels, we can construct the $P(Z/M)_{uu}$ matrix as follows:

$$[P(Z/M)]_{uu} = \begin{array}{c|cccccccc} Z \backslash M & 000 & 001 & 010 & 011 & 100 & 101 & 110 & 111 \\ \hline 000 & q^3 + p_1 & q^2 p & q^2 p & qp^2 & q^2 p & 0 & 0 & 0 \\ 001 & q^2 p & q^3 + p_2 & qp^2 & q^2 p & qp^2 & q^2 p & 0 & 0 \\ 010 & q^2 p & qp^2 & q^3 + p_3 & q^2 p & qp^2 & p^3 & q^2 p & 0 \\ 011 & qp^2 & q^2 p & q^2 p & q^3 & p^3 & qp^2 & qp^2 & q^2 p \\ 100 & q^2 p & qp^2 & qp^2 & p^3 & q^3 & q^2 p & q^2 p & qp^2 \\ 101 & 0 & q^2 p & p^3 & qp^2 & q^2 p & q^3 + p_3 & qp^2 & q^2 p \\ 110 & 0 & 0 & q^2 p & qp^2 & q^2 p & qp^2 & q^3 + p_2 & q^2 p \\ 111 & 0 & 0 & 0 & q^2 p & qp^2 & q^2 p & q^2 p & q^3 + p_1 \end{array} ,$$

where

$$p_1 = 2qp^2 + p^3 ,$$

$$p_2 = qp^2 + p^3 ,$$

and

$$p_3 = qp^2 .$$

The Compressed-Uncoded System

Now for D_{cu} we have the expression:

$$D_{cu} = \overline{(CR)}_m \sum_{M,Z} P(M)_{cu} P(Z/M)_{cu} D(M, Z) ,$$

where $\overline{(CR)}_m$ is the average word compression ratio defined in Chapter IV. D_{cu} has the factor $\overline{(CR)}_m$ because on the average, an error in a compressed data word causes $\overline{(CR)}_m$ words to be in error in the reconstructed data. Now $P(M)_{cu}$ is the first-order probability of the compressed data.

The matrix $P(Z/M)_{cu}$ is the word probability of the entire system, but in this case, the data user has little or no redundancy left in the compressed data or the timing data for use in error control. Therefore, $P(Z/M)_{cu}$ can be written simply as the word probability imposed by the channel. For a 3-bit word system over a binary symmetric channel, $P(Z/M)_{cu}$ would appear exactly as the $[P(Z/M)]$ shown above.

The Compressed-Coded System

The third distortion D_{cc} is given by:

$$D_{cc} = \overline{(CR)}_m \sum_{M,Z} P(M)_{cu} P(Z/M)_{cc} D(M, Z) .$$

Again, the factor $\overline{(CR)}_m$ accounts for the fact that a word in error in compressed, decoded data causes, on the average, $\overline{(CR)}_m$ words to be in error in the final reconstructed data. The compressed, uncoded data is the source for this channel encoding, and thus $P(M)_{cu}$ is used. Now the system transition probability matrix $P(Z/M)_{cc}$ has to take into account the error-control properties of the code.

The error-correction process will provide a situation in which three things can happen to an n -bit transmitted word, assuming an (n, k) block code with k information bits:

1. No errors are introduced and the word is properly decoded.

2. The number of errors is equal to or less than t , and the t -error-correcting code decodes the k -bit words properly.
3. The number of errors is greater than t , and an original k -bit word is decoded into some other k -bit word.

As is shown in Appendix A, we can compute the probability of the first two alternatives above for a Binary Symmetric Channel as:

$$\text{Probability of Correct Decoding} = P_c = q^n + \sum_{j=1}^t \binom{n}{j} q^{n-j} p^j .$$

The probability of the third alternative is then given by:

$$\text{Probability of Incorrect Decoding} = 1 - q^n - \sum_{j=1}^t \binom{n}{j} q^{n-j} p^j .$$

When a k -bit word is improperly decoded into some other k -bit word, some number of errors greater than the t -error-correcting capacity of the code changes bits in either the information or check sections of the code word or in both of these sections. For each particular code, one could examine the effect of all possible error patterns corresponding to more than t errors and compute the probability of a given word z_j being decoded when word m_i ($i \neq j$) was originally encoded. The number of cases one would have to examine would be equal to

$$2^k \sum_{i=t+1}^n \binom{n}{i} .$$

For a (13,9) Hamming code ($t = 1$), for example this number would be

$$2^9 \sum_{i=2}^{13} \binom{13}{i} = 2^9 \left(2^{13} - \sum_{j=0}^1 \binom{13}{j} \right) = 2^9 (2^{13} - 1 - 13) \approx 2^{22} ,$$

or approximately 4.6×10^6 cases. For the purposes of tradeoff analysis, wherein codes of different lengths and different error-correcting capabilities may be compared, it is reasonable to make an approximation and assign equal transition probabilities to the *incorrectly* decoded word pairs. This probability is given by:

$$\text{Probability of Each Incorrectly Decoded Word Pair} = \frac{1 - q^n - \sum_{j=1}^t \binom{n}{j} q^{n-j} p^j}{2^k - 1} = \frac{1 - P_c}{2^k - 1} .$$

Then the $P(Z/M)_{cc}$ matrix is constructed by using the following values for elements:

$$\text{Each main diagonal element} = P_c ,$$

$$\text{Every other element} = \frac{1 - P_c}{2^k - 1} .$$

An interesting verification of the usual statement that "controlled redundancy is better than uncontrolled redundancy" can be obtained by comparing $P(Z/M)_{uu}$ and $P(Z/M)_{cc}$. The truncation of $P(Z/M)_{cu}$ to form $P(Z/M)_{uu}$ eliminates elements of high and low probability values, whereas a single-error-correcting code, for example, eliminates elements of equal and *relatively high* probability values $[(1 - p)^{n-1} p]$, to form $P(Z/M)_{cc}$.

The distortion measure matrix $D(M, Z)$ used in all three D expressions assigns a relative cost to word errors. In general, each element of the $D(M, Z)$ matrix may be any function of the dissimilarity between m_i and z_j . One distortion measure that can be used is the absolute difference between m_i and z_j , or

$$c(m, z) = |z_j - m_i| .$$

Another measure that is widely used in evaluating systems is the squared difference, or

$$c(m, z) = (z_j - m_i)^2 .$$

Before any realistic comparisons can be made, the additional distortion caused by timing errors must be accounted for.

Effect of Time Errors on Distortion

The effect of errors in the time information will be represented as an additional average distortion defined as:

$$D_t = \overline{(CR)}_m \sum_{M, Z} P(M) P(Z/M)_t D(M, Z) ,$$

where $P(Z/M)_t$ is the system transitional word probability of z_j being received given that m_i was sent, which probability is due to time errors alone.

For the purpose of comparing distortions caused by time errors, the uncompressed, uncoded system will be considered free of distortion due to time errors, or $P(Z/M)_{tuu} = [I]$. This assumption is made since in an uncompressed system, the telemetry word sequence has a known pattern

(as described in Chapter II) with synchronization words and frame counts included. With such a timing structure, most time errors can be at least detected, if not also corrected in many cases.

For the compressed-uncoded (CU) and compressed-coded (CC) systems, we consider errors in the time words using the model of a binary-symmetric channel. In both the CU case and the CC case error detection is possible because the time words form a strictly-increasing monotonic sequence. This built-in error detection feature can be described as follows.

Consider a sequence of time words from an unmodified polynomial-predictor type of IP data compressor as described in Chapter II. In this system each time word corresponds to a particular non-redundant data sample. The sequence can be shown pictorially as:

$$T_1 t_1 t_2 t_3 t_4 t_5 \cdots t_k \cdots T_2 ,$$

where T_1 and T_2 are the absolute time words sent periodically so that $T_2 - T_1$ is a constant, and t_1, t_2, \dots are the times of the non-redundant data samples in the time interval T_1 to T_2 , all measured from T_1 . Now, let us assume that an error in transmission causes one of the incremental time words to be changed to an erroneous value in the range 0 to $T_2 - T_1$. Assuming error-free transmission of the absolute time words, there are three different cases to examine.

Case 1: t_k is changed to a value t_e where $t_{k-1} < t_e < t_{k+1}$. In this case no error detection is possible.

Case 2: t_k is changed to a value t_e where a) $t_{k+1} \leq t_e < t_{k+2}$, or b) $t_{k-2} < t_e \leq t_{k-1}$. Now in both a and b a time word is out of order, but there is also an ambiguity as to which time word is in error. In a, t_k or t_{k+1} could be in error. In b, either t_k or t_{k-1} could be in error. So error detection with an ambiguity over two time words is possible.

Case 3: t_k is changed to a value t_e where a) $t_{k+2} \leq t_e$, or b) $t_e \leq t_{k-2}$. Now in both a and b a time word is out of order, but in each situation there is *no* ambiguity as to which time word is in error. The out-of-order time word is in error. So error detection is possible in this case.

We can generalize from the above three cases and say that unambiguous error detection is possible when the error changes the time word to a value equal to or greater than, or equal to or less than, two time words greater than, or less than, respectively, the original word. If we put this on an average basis, the average detectable error is then equal to twice the average adjacent incremental time word difference or

$$\text{Average Detectable Time Error} = 2(\overline{\text{CR}})_m \left(\frac{T_2 - T_1}{m} \right) \text{ in seconds ,}$$

where m is the number of sample time intervals in $T_2 - T_1$.

A time-word transitional probability matrix can be constructed to give the conditional probability of time word t_r being received, given that time word t_s was sent. If there are $m - 1$ possible

incremental time words in $T_2 - T_1$, then each time word can be binary coded into k_t bits, where k_t is an integer and $k_t \geq \log_2 (m - 1)$. Now, since on the average all errors equal to or greater than two average incremental time differences may be detected, the $[P(t_r/t_s)]$ matrix is corner-truncated in a manner similar to the way $P(Z/M)_{uu}$ was obtained, and all elements corresponding to a vertical distance from the main diagonal equal to or greater than $2(\overline{CR})_m [(T_2 - T_1)/m]$ are set to zero. Now in order to satisfy

$$\sum_r P(t_r/t_s) = 1 ,$$

we use the convention of adding the total probability removed from each column by the truncation process to the main diagonal element of the column. This corresponds on the average to changing an unambiguously-detected erroneous time word to a value midway between its neighbors, or $(t_{k+1} + t_{k-1})/2$. Now we examine the individual cases of CU and CC.

For the compressed-uncoded system (CU) we assume that no error-control coding of the time information takes place. Then $[P(t_r/t_s)]$ takes the form exactly as described above where we truncate a matrix for the BSC. Now in order to translate $[P(t_r/t_s)]$ into an equivalent $[P(Z/M)_t]$ for the data values we consider a two-value probability matrix: one value for correct transmission, the other value for incorrect transmission. Then, averaging $P(t_r = t_s)$,

$$P(Z = M)_{t_{cu}} = \overline{P(t_r = t_s)}_{cu} ,$$

$$P(Z \neq M)_{t_{cu}} = \frac{1 - \overline{P(t_r = t_s)}_{cu}}{2^k - 1} ,$$

and

$$D_{t_{cu}} = (\overline{CR})_m \sum_{M, Z} P(M)_{cu} P(Z/M)_{t_{cu}} D(M, Z) .$$

For the compressed-coded (CC) system, we assume that the time words will have error-control coding. Then, since the decoding operation will normally take place before the gross error-detection described above, we construct the matrix $[P(t_r/t_s)]$ in the following way. First, let each element correspond to the word probabilities over a binary symmetric channel. Then make the main diagonal terms correspond to the probability of correct decoding. For a (n_t, k_t) t -error-correcting block code,

$$P_c = \text{Prob. of correct decoding} = q^{n_t} + \sum_{j=1}^{t} \binom{n_t}{j} q^{n-j} p^j .$$

Make all other elements equal to $(1 - P_c) / (2^{k_t} - 1)$. Now, truncate the matrix as in the CU case and add the removed probability to the main diagonal. This corresponds to an addition of

$$\left((m-1) - 2\overline{(\text{CR})}_m \right) \left(\frac{1 - P_c}{2^{k_t} - 1} \right)$$

for the first column, etc. Then, as in the case of CU,

$$P(Z = M)_{tcc} = \overline{P(t_r = t_s)}_{cc} ,$$

$$P(Z \neq M)_{tcc} = \frac{1 - \overline{P(t_r = t_s)}_{cc}}{2^k - 1} ,$$

and

$$D_{tcc} = \overline{(\text{CR})}_m \sum_{M, Z} P(M)_{cu} P(Z/M)_{tcc} D(M, Z) .$$

The above analysis has been for the unmodified algorithms described in Chapter II. As discussed in Chapter IV, there are also the modified algorithms and the sequence coding methods of time encoding. For the present purpose of obtaining a quantitative measure of distortion due to time errors, the sequence coding method will not be considered since, as was pointed out in Chapter IV, the greater overall effect of errors in its code plus its greater inherent hardware complexity render it undesirable in most cases.

Now, errors in the modified algorithm codes will produce an equal or greater distortion than in the case of the unmodified algorithms. Following the notation of Chapter IV, when $k = 1$, the distortions due to code errors in the modified and unmodified algorithms are the same. As k decreases and the non-redundant samples are less equally spaced throughout the data stream, the distortion corresponding to the modified algorithm becomes larger than that of the unmodified algorithm. In the limit, when k reaches its minimum value and all non-redundant samples in the absolute time interval $T_2 - T_1$ are contiguous, then a time error in the single time word at the beginning of the non-redundant block could make all $(m - 1)$ samples in error in the $T_2 - T_1$ interval. In this case, we would multiply the summations in the D_t expressions by $(m - 1)$ instead of $\overline{(\text{CR})}_m$.

Chapter VI

USE OF THE TRADEOFF MEASURES

In this chapter, the use of the measures of rate-ratio and distortion developed in the previous chapter will be described and the effect of some system parameters on these measures will be examined. A logical procedure for choosing various systems through the use of these measures will be given. The effect of the channel on the distortion will be examined using the binary symmetric channel model. Tradeoffs between distortion and rate-ratio will be examined as functions of the compression ratio. The effect of the source probability distribution on the distortion will be examined, and finally a graphic representation of the various tradeoffs will be given.

Throughout this chapter, the same three systems as used in the previous chapter will be used with the indicated abbreviations in subscripts, etc.: uncompressed-uncoded (UU); compressed-uncoded (CU); and compressed-coded (CC).

A Rationale for Choosing a System

In order to compare space telemetry systems with different combinations of compression and coding, the following quantities are calculated for each system:

$$\text{For UU: } R_{uu} = 1 \qquad D_{suu} = D_{uu} ,$$

$$\text{For CU: } R_{cu} \qquad D_{scu} = D_{cu} + D_{tcu} ,$$

$$\text{For CC: } R_{cc} \qquad D_{scc} = D_{cc} + D_{tcc} ,$$

where D_s stands for the system distortion due to data errors *and* time errors.

Then for each data source, the maximum allowable average distortion D_{max} is obtained from the data user for a given distortion measure $D(M, Z)$. The various distortions and rate-ratios are then compared, with the following criterion in mind: An acceptable system is one whose distortion is less than D_{max} and whose rate-ratio is smaller than R_{uu} .

The procedure of comparing R's and D's and choosing systems can best be described by the use of a flow chart (Figure 10) where the assumption has been made that:

$$D_{suu} < D_{scu} > D_{scc} .$$

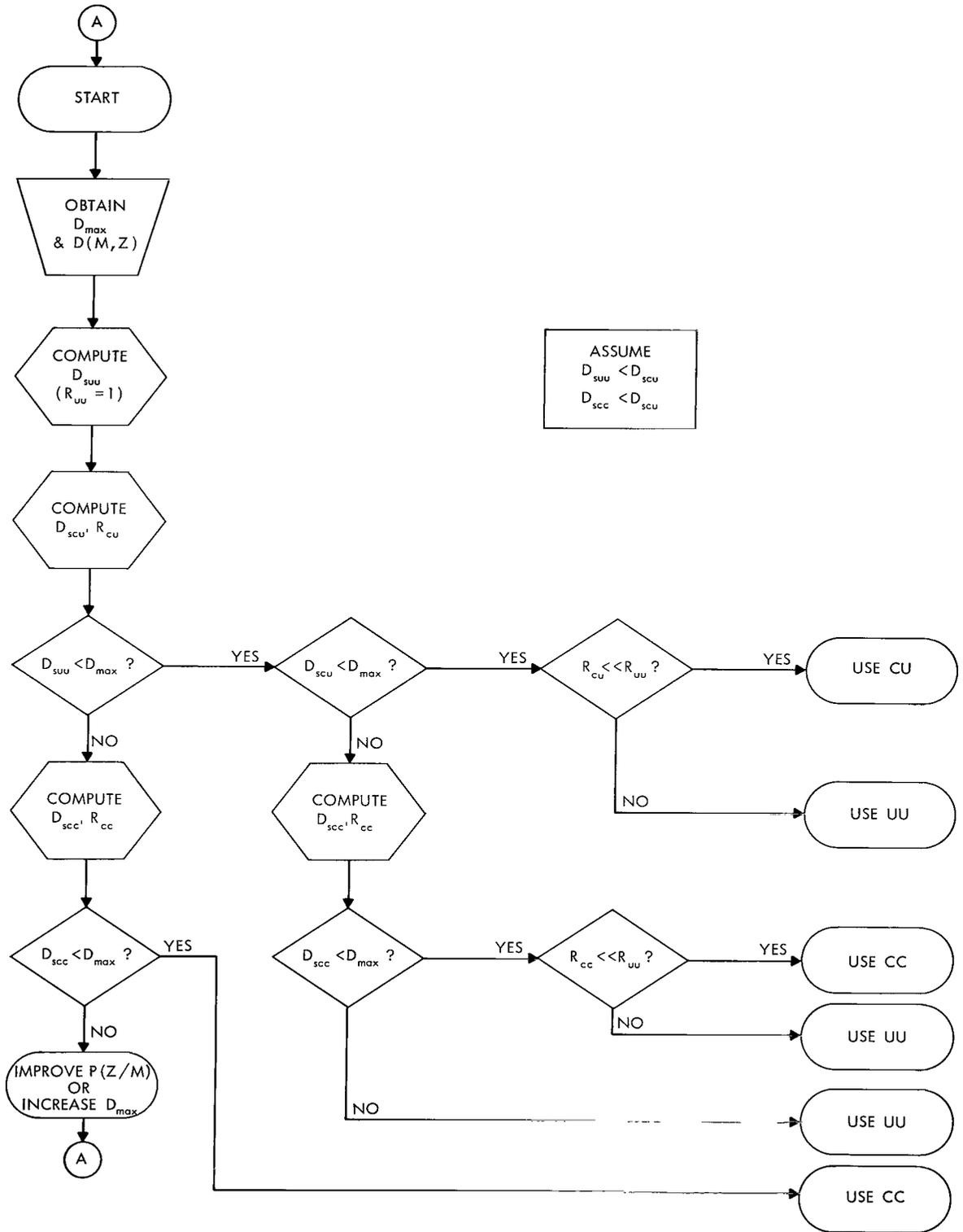


Figure 10—Flow chart of rationale for making system decisions among UU, CU, and CC.

The above inequalities are based on three conditions: first, error-control coding reduces distortion; second, use of inherent redundancy in uncompressed data to control errors also reduces distortion; and third, errors in compressed data are expanded in reconstructed data.

As indicated in Figure 10, the $D(M, Z)$ matrix and the maximum allowable average distortion D_{max} are obtained—typically from the experimenter. The underlying philosophy of the succeeding tests is to find the simplest system that will achieve an appreciable reduction in rate-ratio and yet not exceed the specified maximum average distortion.

Now the choices shown in the flow chart can be explained in order (top to bottom). The first test determines if the CU system will be the choice. If the CU distortion is less than D_{max} and an appreciable reduction in rate-ratio is achieved, *then the CU system is chosen*. If the reduction in rate-ratio is not appreciable and the UU system meets the D_{max} requirement, *then the UU system is chosen*, which means there is no compression or coding in the system. If CU does not meet the D_{max} requirement, it may be possible to code and have the CC system meet this requirement and yet realize an appreciable reduction in rate-ratio. In this latter case *the CC system is chosen*. However, if the coding does not achieve the rate-ratio reduction desired, and the UU system still meets the D_{max} requirement, *then the UU system is chosen*. Also *the UU system is chosen* if the coding does not meet the D_{max} requirement. As can be seen from the flow chart, the above tests and choices are made under the condition that the UU system meets the D_{max} requirement and because of this, we always fall back on the UU system when the other systems fail either the rate-ratio or distortion test.

Now, under the condition that the UU system does not meet the D_{max} requirement, then the *CC system is tried*. If the CC system does not meet the D_{max} requirement under these conditions, then none of the three systems (UU, CU, and CC) can be chosen.

The flow chart shows that there are three ways of coming to a choice of system UU, two ways to system CC, one way to system CU, and finally one way to none of them. In the last case none of the three systems meets the distortion requirements, and a new comparison must be made with an improved $P(Z/M)$ or increased D_{max} . An improvement in the $P(Z/M)$ of all three systems can be obtained by a channel improvement, and an improvement in $P(Z/M)_{cc}$ can be obtained by a code that corrects more errors.

One may look at the above procedure as a joint minimization of R and D within given bounds, and this concept will be graphically illustrated below.

Effect of the Channel - An Example

The average distortion measures developed in the previous chapter involved a $P(Z/M)$ matrix which was determined by a number of factors. In the case of a binary-symmetric channel model, a function of the bit-error probability constitutes each element of $P(Z/M)$. In the case of the uncompressed-uncoded (UU) system, the $P(Z/M)$ matrix is modified to take into account the ability of the data user to spot errors using the inherent data redundancy. In the case of the

compressed-coded (CC) system, the $P(Z/M)$ matrix is formed in accordance with the number of errors the code corrects. The following example illustrates the effect of the channel.

Now in order to show the effect of just the channel on the average distortion, we plot the ratio of average distortion to word compression ratio. We also plot the average distortion in the UU system for *no* error-control on the part of the data user. A uniform source probability distribution is used throughout for the plots, and the coding is as follows: for data, a (6,3) Hamming single-

error-correcting code for 3-bit data information word; and for time, a (7,4) Hamming single-error-correcting code for a 4-bit time information word.

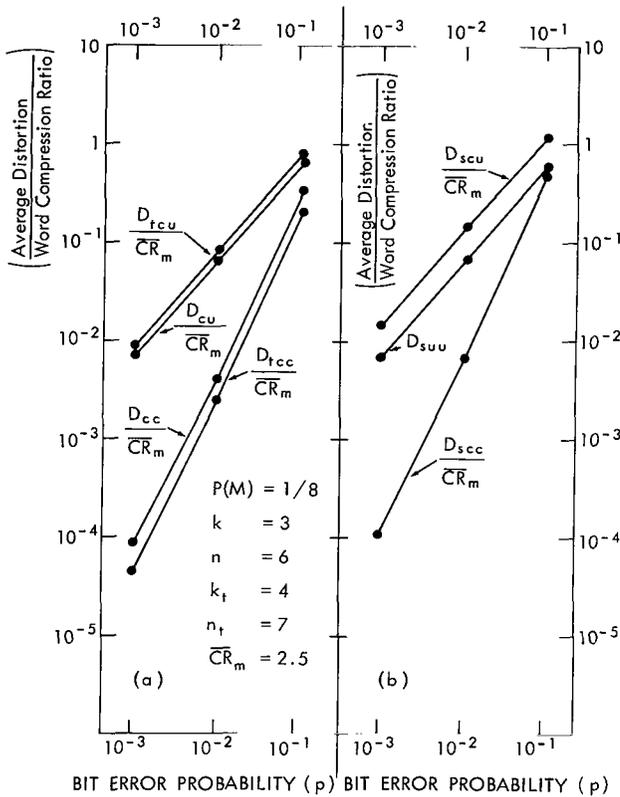


Figure 11—Ratios of distortion to word compression ratio for different values of p over a BSC.

The compression-ratio-normalized average distortions due to data errors, D_{cu}/\overline{CR}_m and D_{cc}/\overline{CR}_m , as well as those due to time errors, $D_{t_{cu}}/\overline{CR}_m$ and $D_{t_{cc}}/\overline{CR}_m$, are plotted in Figure 11a. An interesting characteristic of these plots is that in both the CU and CC systems, the normalized distortion due to time errors is within the same order of magnitude as the normalized distortion due to data errors. The total normalized distortions for the CU and CC systems as well as the distortion for the UU system are plotted in Figure 11b. It can be seen from these plots that the relative improvement in performance (less normalized distortion) achieved by coding depends on the channel error probability, with less improvement at larger bit-error probabilities. By contrast, the *relative* degradation in performance due to compression $D_{s_{cu}}/\overline{CR}_m$ is independent of the channel bit-error probability.

The normalized values of $D_{t_{cu}}$ and $D_{t_{cc}}$ shown in Figure 11 involve, respectively, $P(Z/M)_{t_{cu}}$ and $P(Z/M)_{t_{cc}}$ matrices that were computed for a word compression ratio of 2.5. Higher values of word compression ratio would increase the normalized values of $D_{t_{cu}}$ and $D_{t_{cc}}$ in Figure 11. More will be said about this particular effect of the word compression ratio in the next section.

Effect of the Compression Ratio

Since the expressions for average distortion for the compressed-uncoded (CU) and compressed-coded (CC) systems are both functions of the word compression ratio \overline{CR}_m , the specification of a maximum allowable distortion sets an upper bound on \overline{CR}_m . On the other hand, the rate-ratio R , in

bits per uncompressed bits, is a function of the reciprocal of the actual average compression ratio \overline{CR}_{m_a} which in turn is a function of \overline{CR}_m . (See Figure 9 for a plot of \overline{CR}_{m_a} vs. \overline{CR}_m for a typical system.) So the requirement of a worthwhile reduction in rate-ratio sets a lower bound on \overline{CR}_m . There are situations in which these upper and lower bounds will form a region of acceptable values of \overline{CR}_m . However, in those cases where the upper bound falls below the lower bound, no value of \overline{CR}_m is acceptable, and the system in question, either CU or CC, will not meet the D and R specifications. This concept of an acceptable *range* of values of word compression ratio, rather than the idea that high word compression ratio is always desirable, is an important outcome of the application of distortion and rate-ratio measures to a data-compressed system. Examples of different situations involving acceptable ranges of values of the word compression ratio are shown by the plots in Figure 12.

The curvature of the $D_{s_{cu}}$ and $D_{s_{cc}}$ lines in Figure 12 comes about because of the double effect of the word compression ratio \overline{CR}_m in the expressions for $D_{t_{cu}}$ and $D_{t_{cc}}$. As was shown in Chapter V, the computations of both $D_{t_{cu}}$ and $D_{t_{cc}}$ involve a corner truncation of a matrix $P(t_r/t_s)$ in accordance with the error-control property inherent in the strict monotonicity of the time words. The amount of truncation depends upon the word compression ratio, viz. the higher the compression ratio, the less truncation. This relationship is a linearly decreasing one for each column of $P(t_r/t_s)$. For example, in the first column of $P(t_r/t_s)_{cu}$, the number of elements set to zero by the truncation is $m - 1 - 2(\overline{CR}_m)$. The sum of the probabilities of the truncated elements in each column is added to the main diagonal element of $P(t_r/t_s)_{cu}$. The main diagonal elements $P(Z = M)_{t_{cu}}$ of the resulting $P(Z/M)_{t_{cu}}$ matrix are set equal to the average of the main diagonal elements of $P(t_r/t_s)_{cu}$, and all off-diagonal elements of $P(Z/M)_{t_{cu}}$ are set equal to the same fractional part of $1 - P(Z = M)_{t_{cu}}$. So \overline{CR}_m finally affects the $P(Z/M)_{t_{cu}}$ matrix in a linear way. When we compute

$$D_{t_{cu}} = \overline{CR}_m \sum_{M, Z} P(M)_{cu} P(Z/M)_{t_{cu}} D(M, Z) ,$$

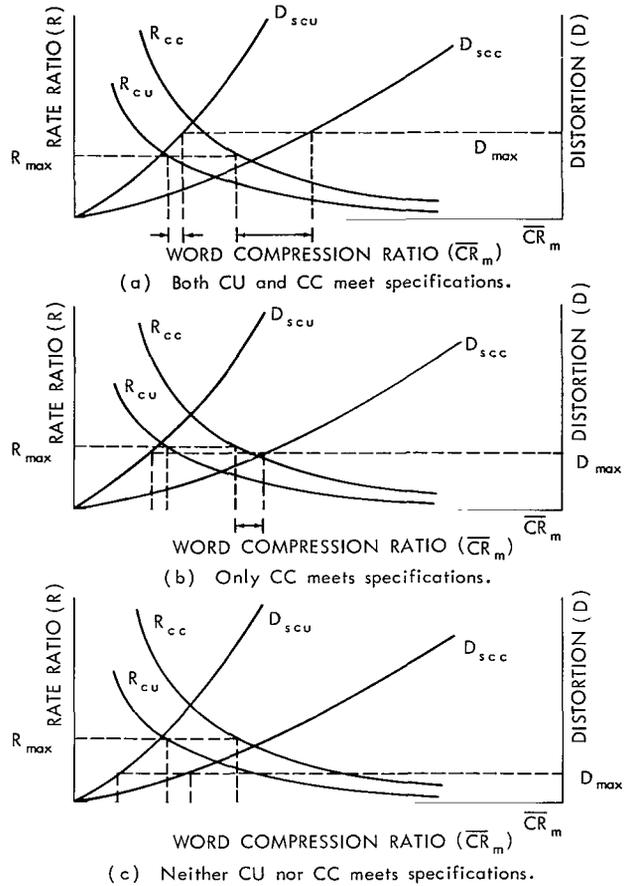


Figure 12—Effect of D and R specifications on the range of choice of the word compression ratio.

$(\overline{CR}_m)^2$ will appear in the calculation, but with an *increasing* effect on $D_{t_{cu}}$. The same description applies to $D_{t_{cc}}$.

Effect of the Source Probability Distribution

In the computation of average distortion for the systems UU, CU and CC, the probability distribution of the source in question $P(M)$ enters as a weighting on a summation as follows:

$$\frac{D}{\overline{CR}_m} = \sum_{M, Z} P(M) P(Z/M) D(M, Z) ,$$

(where in the UU case, $\overline{CR}_m \equiv 1$)

$$\frac{D}{\overline{CR}_m} = \sum_M \left[P(M) \sum_Z P(Z/M) D(M, Z) \right] .$$

Now, when the M set and Z set are identical sets of data levels (which is the usual case in space telemetry) we can note symmetrical characteristics in the expression

$$\sum_Z P(Z/M) D(M, Z)$$

as a function of M. Consider the M set to be made up of two ranges of levels for a k-bit data word:

Lower range of levels: (0) to $(2^{k-1} - 1)$

Upper range of levels: (2^{k-1}) to $(2^k - 1)$

Now $P(Z/M)_{cu}$ has symmetry around its two diagonals since it is just the transitional word probability of a binary-symmetric channel. The matrix $P(Z/M)_{uu}$ also has symmetry around its two diagonals since it was obtained by setting a symmetrical set of diagonals parallel to the main diagonal in $P(Z/M)_{cu}$ equal to zero, and then adding sums of deleted column elements to the main diagonal. $P(Z/M)_{cc}$ also has two-diagonal symmetry since all off-diagonal terms were set equal to the same fractional part of the probability of incorrect decoding and all main diagonal terms are equal. Finally $D(M, Z)$ has two-diagonal symmetry since each element is the same function of the discrepancy between m_i and z_j .

Then the product of any one of the above $P(Z/M)$ matrices and the $D(M, Z)$ matrix will result in another matrix with symmetry about the two diagonals. Now this two-diagonal symmetry gives rise to symmetry between the upper and lower level ranges for the sums

$$\sum_Z P(Z/M) D(M, Z) .$$

For the binary symmetric channel model and the absolute difference distortion measure $c(m, z) = |z_j - m_i|$, the sums

$$\sum_Z P(Z/M) D(M, Z)$$

have a symmetrical concave upward shape as a function of data level. With this general shape in mind, we can examine the effect of various $P(M)$ distributions on the final value of average distortion.

For the parameters indicated therein, the values of D/\overline{CR}_m are plotted* in Figure 13 for three different symmetrical shapes of $P(M)$: concave upward, concave downward and uniform. It is to be noted that the uniform case also covers the cases where the $P(M)$ distribution consists of uniform halves ($P(M)_{lower}$ and $P(M)_{upper}$). Since

$$\sum_M P(M) = 1,$$

any one of these distributions can be made equivalent to the uniform distribution (as far as D is concerned) by merely folding the distribution about the center of the data level range, adding the two $P(M)$ values and computing D as:

$$D = (\overline{CR})_m \sum_{M(lower)} \left\{ \left[P(M)_{lower} + P(M)_{upper} \right] \sum_Z P(Z/M) D(M, Z) \right\}.$$

The equivalence to the uniform case is obvious since

$$P(M)_{uniform} = \frac{P(M)_{lower} + P(M)_{upper}}{2}.$$

Figure 13 shows relatively small differences between the D 's for the three $P(M)$ distributions. As would be expected, the concave upward $P(M)$ distribution gives the largest value of D because of

*With the simplifying assumption that $P(Z/M)_{uu} = P(Z/M)_{cu}$

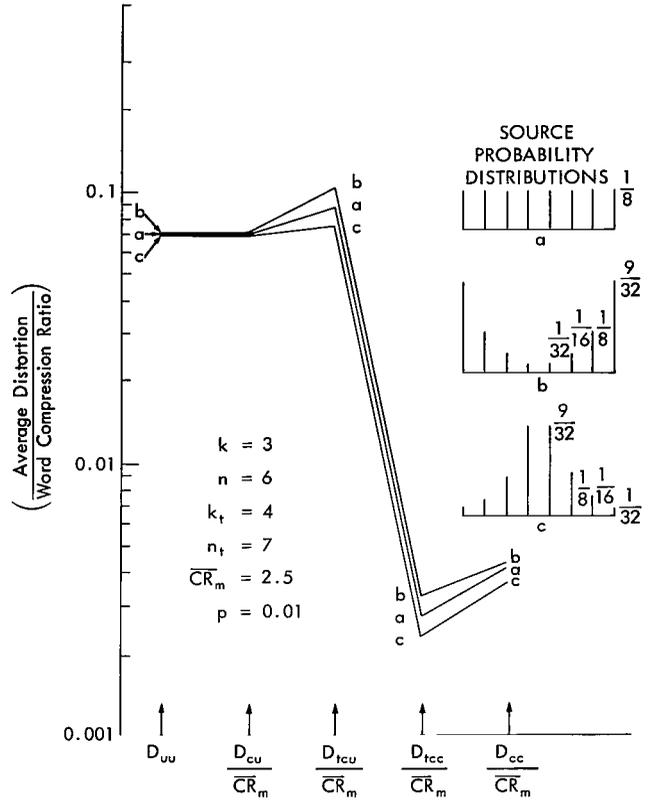


Figure 13—Ratios of distortion to word compression ratio for three different source probability distributions.

the concave upward shape of the sum

$$\sum_z P(Z/M) D(M, Z) .$$

In order to examine the effect of differently shaped $P(M)$'s on D as the word length k increases, we can say the following: Basically there are two types of sums

$$\sum_z P(Z/M) D(M, Z)$$

involved in D by nature of the structure of $P(Z/M)$. (The matrix $D(M, Z)$ is the same for all sums.) One type has all main diagonal elements of $P(Z/M)$ equal to one probability, and all off-main-diagonal elements equal to a different probability. The distortions involved with this type of sum are D_{cc} , D_{tcc} and D_{tcu} . The second type has many elements in the matrix $P(Z/M)$ with the property of symmetry about both diagonals. This type includes D_{uu} and D_{cu} .

For the first type above, the concave upward shape is determined by the sums of the columns of $D(M, Z)$ since the main diagonal terms of $P(Z/M)$ are multiplied by the zeros of the main diagonal terms of $D(M, Z)$. This leaves the off-diagonal element value of $P(Z/M)$ multiplied by each non-zero element of $D(M, Z)$. If we consider a k -bit data word, then there is an even number of data levels in M , and we can examine one half of the range since there is symmetry in

$$\sum_z P(Z/M) D(M, Z)$$

as a function of M . For the difference distortion measure of $c(m, z) = |z_j - m_i|$, the end columns of $D(M, Z)$ are each a single arithmetic progression starting with zero and ending with $2^k - 1$. The middle two columns of $D(M, Z)$ are each made up of two arithmetic progressions, one going from zero to $2^{k-1} - 1$, the other going from one to 2^{k-1} . We can compute the ratio of the sum of an end column to the sum of a middle column as follows:

Using the arithmetic progression sum formula:

$$\text{Sum of terms} = \frac{\text{No. of terms}}{2} (\text{first term} + \text{last term}) .$$

For the end column above:

$$\text{Sum} = \frac{2^k}{2} (0 + 2^k - 1) .$$

For the middle column:

$$\text{Sum} = \frac{2^{k-1}}{2} (0 + 2^{k-1} - 1) + \frac{2^{k-1}}{2} (1 + 2^{k-1}) .$$

Then the ratio of the sums of the end and middle columns is

$$\frac{\text{End Sum}}{\text{Middle Sum}} = \frac{2^k - 1}{2^{k-1}} .$$

As k becomes infinite,

$$\lim_{k \rightarrow \infty} \frac{2^k - 1}{2^{k-1}} = \lim_{k \rightarrow \infty} \left(2 - \frac{1}{2^{k-1}} \right) = 2 .$$

So the concave shape does *not* grow more exaggerated as the data word length increases. For the eight-level case used for the plots of Figure 13, the ratio corresponding to that given above is

$$\frac{\text{End Sum}}{\text{Middle Sum}} \Big|_{k=3} = \frac{7}{4} = 1.75 .$$

Thus the results shown in Figure 13 for D_{cc} , D_{tcc} and D_{tcu} are meaningful for cases where the data word is longer than 3 bits.

Similarly, for the difference distortion measure of $c(m, z) = (z_j - m_i)^2$ we can calculate a similar limit as above. In this case the end columns of $D(M, Z)$ are each sequences of squared integers starting with zero and ending with $(2^k - 1)^2$. The middle two columns of $D(M, Z)$ are each made up of two sequences of squared integers, one going from zero to $(2^{k-1} - 1)^2$, the other going from one to $(2^{k-1})^2$.

Using the formula for the sum of squared integers with highest integer n :

$$\text{Sum} = \frac{n(n+1)(2n+1)}{6} ,$$

For the end column: $n = 2^k - 1$,

$$\text{Sum} = \frac{(2^k - 1)(2^k)(2^{k+1} - 1)}{6} .$$

For each middle column: $n_1 = 2^{k-1}$, $n_2 = 2^{k-1} - 1$

$$\text{Sum} = \frac{n_1(n_1+1)(2n_1+1) + n_2(n_2+1)(2n_2+1)}{6} .$$

Then the ratio of the sums of the end and middle columns is:

$$\frac{\text{End Sum}}{\text{Middle Sum}} = \frac{\left(1 - \frac{1}{2^k}\right) \left(2 - \frac{1}{2^k}\right)}{\frac{1}{2} \left(\frac{1}{2} + \frac{1}{2^k}\right) \left(1 + \frac{1}{2^k}\right) + \left(\frac{1}{2} - \frac{1}{2^k}\right) \left(\frac{1}{2}\right) \left(1 - \frac{1}{2^k}\right)}$$

As k becomes infinite

$$\lim_{k \rightarrow \infty} \frac{\text{End Sum}}{\text{Middle Sum}} = 4 .$$

For the eight-level system with $c(m, z) = (z_j - m_i)^2$, the ratio corresponding to that given above is

$$\frac{\text{End Sum}}{\text{Middle Sum}} \Big|_{k=3} = \frac{140}{44} = 3.18 .$$

For the D_{uu} and D_{cu} sums, actual calculations show that the 3-bit cases produce sums

$$\sum_z P(Z/M) D(M, Z) ,$$

which are not as *concave* as those for D_{cc} , D_{tcc} and D_{tcu} . This "flattening out" effect can be explained by the fact that here the weighting imposed by $P(Z/M)$ on each $D(M, Z)$ is a fluctuating, rather than a smoothly varying, function of Z . It is to be expected that the same effect will hold as k is increased since $P(Z/M)_{uu}$ and $P(Z/M)_{cu}$ will still contain elements that are word probabilities over a binary symmetric channel, and the values of these elements will fluctuate over any row or column, with the sum in each case equal to unity.

The conclusion to be drawn from the above analysis is that the shape of the source probability distribution, either $P(M)_{uu}$ or $P(M)_{cu}$, does not have a critical effect on the average distortion.

Graphical Representation of Tradeoffs - An Example

In the process of choosing compression and coding schemes for space telemetry, one can portray the tradeoffs between data compression efficiency and data reliability by means of a rate-ratio and distortion diagram similar to that used by Shannon for the rate distortion function (Reference 12). In this diagram, we plot the values of R and D_s for the three systems UU, CU and CC, for different values of bit-error probability p , over a binary symmetric channel. We connect the points corresponding to one value of p by straight lines with arrows showing the progression from system UU to system CU and finally to system CC. The maximum allowable distortion, as well as the desired overall rate-ratio, can be conveniently drawn on this diagram. Therefore, on one diagram, we can see the possible choices and the relative advantages or disadvantages of these choices in the system design.

Such a diagram is shown in Figure 14 for the example described therein. The distortions for the UU system with different amounts of error correction by the data user utilizing the natural redundancy in the data are shown in this figure. The compression ratios are calculated in the following way:

In order to plot the R's and D's for the three types of systems for different channels we can make $R_{cc} = 1$ and proceed as follows:

$$R_{cc} = 1,$$

$$K_c = \frac{n + n_t}{k + k_t},$$

then

$$\overline{(CR)}_{ma} = K_c.$$

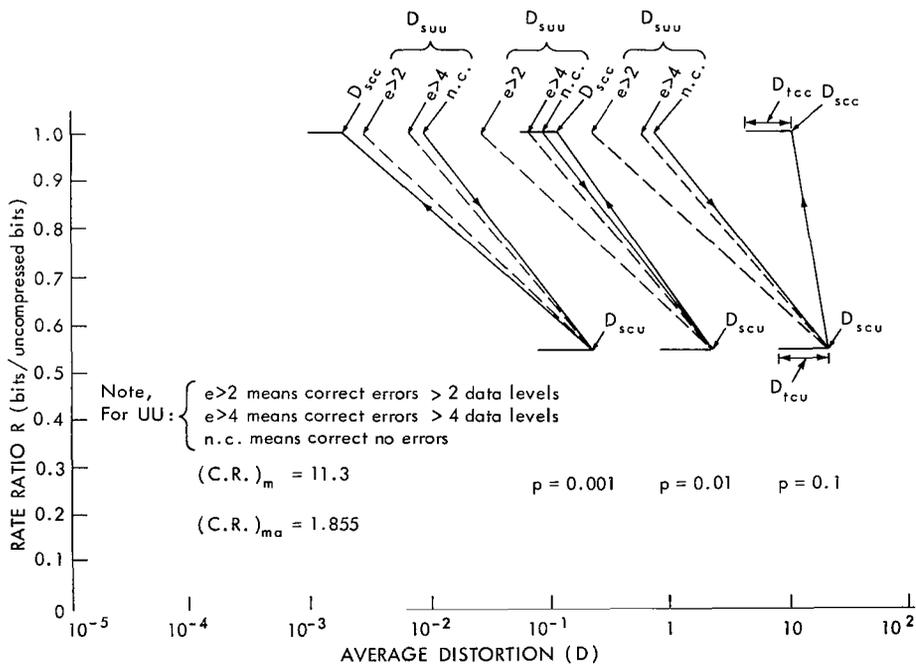


Figure 14—Distortion comparisons for a source with 3-bit data words, 4-bit time words, compressed and coded with (6,3) and (7,4) codes respectively, with $R_{cc} = R_{uu} = 1$.

For the present example, we compute the relation between $\overline{(\text{CR})}_{ma}$ and $\overline{(\text{CR})}_m$ from Chapter IV: (consider time method a , unmodified)

$$\overline{(\text{CR})}_{ma} = \frac{a^2}{\left(2a + s + \frac{a^2 - 2a - s}{\overline{(\text{CR})}_m}\right)} \left(1 + \frac{\log_2 a}{k}\right).$$

Let $a = 16$, $s = 8$, and $k = 3$.

$$\overline{(\text{CR})}_{ma} = \frac{256}{\left(40 + \frac{216}{\overline{(\text{CR})}_m}\right)} \frac{7}{3}.$$

Rearranging,

$$\overline{(\text{CR})}_m = \frac{504 \overline{(\text{CR})}_{ma}}{256 - 93 \overline{(\text{CR})}_{ma}}.$$

Now

$$\overline{(\text{CR})}_{ma} = K_c = \frac{13}{7} = 1.855,$$

$$\overline{(\text{CR})}_m = 11.3.$$

This means that we must attain an average word compression ratio of 11.3 in order to make the compressed-coded rate-ratio the same as the uncompressed-uncoded rate-ratio. Plots of R and D using the above numbers are shown in Figure 14.

This example was chosen purposely to show the effect of compression and coding on the distortion only—leaving the rate-ratio unchanged from the UU system rate-ratio. Now as can be seen from the figure, as the channel improves (lower p), greater reductions in distortion can be obtained. It is interesting to note that for this particular example under poor channel conditions, coding does not accomplish much distortion reduction. At $p = 0.1$, $D_{suu} < D_{scc}$, and at $p = 0.01$, we still have $D_{suu} < D_{scc}$. Only at $p = 0.001$ does the coding start to provide lower distortions than those obtained by using the natural redundancy in the data.

Two cases of the use of natural redundancy are shown in Figure 14. In the first case, it is assumed that absolute errors greater than 4 data levels can be detected and corrected in accordance with the matrix truncation procedure described in Chapter V. In the second case, absolute errors greater than 2 levels are assumed to be detected and corrected. The resulting distortions for these cases can be compared to the case of no error detection or correction in the uncompressed uncoded case also shown in Figure 14.

In order to obtain both distortion reduction *and* rate-ratio reduction for the example in Figure 14 one would increase the time between periodic time words, thus increasing the effective main frame size and thereby improving the relationship between \overline{CR}_{ma} and \overline{CR}_m (Figure 9).

Chapter VII

CONCLUSIONS

The aim of this research was to study the considerations involved in the combination of data compression and error-control coding in a space telemetry system.

The overall conclusions are that it is possible to quantitatively compare various combinations of compression and coding on the basis of compression ratio and data reliability, and that it is not always true that the replacement of natural data redundancy with coded redundancy results in improved system performance.

The rate-ratio R and average distortion D , based on parameters used by Shannon in his rate distortion function, were found to constitute a useful measure of performance. The particular advantage of this R and D measure is that it simultaneously accounts for a variety of system parameters and conditions which are sometimes treated separately in systems comparisons. Some of the system parameters of particular interest here were error control by inherent data redundancy, error control by a t -error-correcting code, effect of data source probability distribution, time-encoding cost in terms of the compression ratio, the effect of errors in the compressed-data time information on the final reconstructed data, and the relative costs of errors in space telemetry systems. The present research examined the above parameters in three systems: uncompressed-uncoded, compressed-uncoded and compressed-coded. It was shown that the R and D measure could provide the relative tradeoffs between data compression and overall data distortion for different channel error probabilities and different source probability distributions. In the process of studying the interactions of the above parameters in the tradeoff analysis, some interesting related results were obtained as follows.

The error-control ability of a data user using natural data redundancy can be approximated by a simple modification of the overall telemetry system transition probability matrix; this modification allows a comparison to the error-control ability of a t -error-correcting code.

For a given system, the performance measures developed here can graphically portray the superiority of natural redundancy over coded redundancy for error control as the channel degrades.

Although it is not possible to use the facts that the entropy is decreased in ER compression and increased in IP compression to predict the effect of compression on the source probability distribution, the effect of this distribution is not critical in the average distortion performance measure when used with an absolute error criterion.

A new time encoding scheme for compressed data is developed here which generally is more efficient than the more standard methods, and for some types of data behavior is more efficient than sequence encoding.

An unambiguous encoding and decoding method for sequence—coding the time information of the compressed data is developed here which takes advantage of the strict monotonicity of the time information.

The usual expression for maximum compression ratio is found to be unsuited to the polynomial-predictor class of IP compressors, and a new expression for CR is developed using the concept of compressible sequences from an ergodic information source.

The dual requirements of a maximum allowable distortion and a minimum overall compression ratio can place bounds on the word compression ratio, and in some cases these requirements can be shown to be incompatible.

The distortion effect of errors in the time information for a compressed system can be formulated including the error-control property of monotonicity in the time words.

Goddard Space Flight Center
National Aeronautics and Space Administration
Greenbelt, Maryland, July 11, 1966
125-22-01-17-51

REFERENCES

1. Campanella, S. J., "A Survey of Speech Bandwidth Compression Techniques," *IRE Trans. on Audio* AU-6:104-115, September-October 1958.
2. Cherry, C., Kubba, M. H., Pearson, D. E., and Barton, M. P., "An Experimental Study of the Possible Bandwidth Compression of Visual Image Signals," *Proc. IEEE* 51(11):1507-1517, November 1963.
3. Lynch, T. J., "Space Data Handling at Goddard Space Flight Center," NASA TM X-55180, February 1965.
4. Dudley, H., "Remaking Speech," *J. Acoust. Soc. Amer.* II: 169-177, October 1939.
5. Blasbalg, H., and Van Blerkom, R., "Message Compression," *IRE Trans. on Space Elect. and Telemetry* 8:228-238, September 1962.
6. Davisson, L. D., "Theory of Adaptive Data Compression," In: "Advances in Communication Systems," Volume II, A.V. Balakrishnan, ed., New York: Academic Press, 1966.
7. Ellersick, F. W., "Data Compactors for Space Vehicles," *IRE East Coast Conference on Aerospace and Navigational Electronics Proceedings*, Baltimore, October 23-25, 1961.

8. Gardenhire, L., "Redundancy Reduction—The Key to Active Telemetry," *Proc. National Telemetry Conference*, Los Angeles, 1964.
9. Medlin, J. E., "Sampled Data Prediction for Telemetry Bandwidth Compression," *IEEE Trans. on Space Elect. and Telemetry* 11(1):29-36, March 1965.
10. Fano, R. M., "Transmission of Information," New York: MIT Press and John Wiley and Sons, 1961.
11. Huffman, D. A., "A Method for the Construction of Minimum Redundancy Codes," *Proc. IRE* 40:1098-1101, September 1952.
12. Shannon, C. E., "Coding Theorems for Discrete Sources with a Fidelity Criterion," In: *Information and Decision Processes*, (Robert E. Machol, ed.) pp. 93-126, New York: McGraw-Hill, 1960.
13. "Final Report PCM Data Compression Study, Phase I," Lockheed Missiles and Space Co., Report M-62-65-2, Sunnyvale, California, October 1965.
14. Shannon, C. E., and Weaver, W., "The Mathematical Theory of Communication," Urbana, Ill.: University of Illinois Press, 1949.
15. Hamming, R. W., "Error-Detecting and Error-Correcting Codes," *Bell Sys. Tech. J.* 29:147-160, April 1950.
16. Bose, R. C., and Ray-Chaudhuri, D. K., "A Class of Error-Correcting Binary Group Codes," *Information and Control* 3:68-79, 1960.
17. Bose, R. C., and Ray-Chaudhuri, D. K., "Further Results on Error-Correcting Binary Group Codes," *Information and Control* 3:279-290, September 1960.
18. Peterson, W. W., "Error-Correcting Codes," Cambridge: MIT Press, 1961.
19. Medlin, J. E., "Buffer Length Requirements for a Telemetry Data Compressor," *Proc. National Telemetry Conference*, Paper 14-5, Washington, D. C., 1962.
20. Simpson, R. S., "Buffer Control in Data Compression Systems for Non-Stationary Data," *Proc. National Telemetry Conference*, Los Angeles, 1964.
21. Ash, R., "Information Theory," New York: John Wiley and Sons, 1965.
22. Schwartz, L. S., "Principles of Coding, Filtering, and Information Theory," Baltimore: Spartan Books, 1963.
23. Salzer, H. E., *Tables of $n!$ and $\Gamma(n + 1/2)$ for the First Thousand Values of n* . NBS Applied Math. Series-16, 1951.
24. Gordon, G. A., "An Algorithmic Code for Monotonic Sources," In: *Final Report of GSFC Summer Workshop 1965*, NASA TM X-55356, June 15 - September 15, 1965.

Appendix A

Coding Matrices and Error Probabilities for the Hamming(13,9) Code

Code Characteristics

Since many PCM satellite telemetry systems use 9-bit data words, it is useful to examine a basic error control code for this size word. The Hamming code[†] is a well-known linear code* that can be used for single-error correction, and it provides for a useful numerical comparison of typical coded to uncoded satellite telemetry systems.

It should be noted that for the code chosen, we have as many quantized levels available (2^9) as in the uncoded case. To correct a single error in a binary-symmetric channel, the check digits must be sufficient in number to check one of $n + 1$ happenings: if an error has occurred, and if so in which of the n positions. If the n digits are made up of k information digits and $n - k$ check digits, then

$$2^{(n-k)} \geq (n + 1) .$$

For the case of the code chosen,

$$2^{13-9} = 16 \geq 13 + 1 .$$

Coding Matrices

Let us take a (13,9) Hamming code and start by writing its parity-check matrix simply as follows:[‡]

$$\begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} .$$

[‡]The notation for coding operations used in this appendix is a widely used one, but some symbols inadvertently overlap with some already used in the above chapters. All symbols used in this appendix are defined here.

[†]Hamming, R. W., "Error-Detecting and Error-Correcting Codes," *Bell Sys. Tech. J.* 29, 147-160, 1950.

^{*}Peterson, W. W., "Error-Correcting Codes," Cambridge: MIT Press, 1961.

Reordering:

$$\begin{bmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & | & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & | & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & | & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & | & 0 & 0 & 0 & 1 \end{bmatrix} = P .$$

We consider the encoding operation as a matrix multiplication as follows:

$$[y] = [H'] [x] ,$$

where x is the uncoded word, y is the coded word, and H' is the encoding matrix.

The decoding operation can be represented as follows:

$$[w] = [H] [z] ,$$

where z is the input to the decoder, w is the decoded word, and H is the decoding matrix.

The matrix H' is related to the parity-check matrix as follows:

$$H' = \left[\begin{array}{c|c} I & O \\ \hline A_1 & A_2 \end{array} \right] ,$$

where

$$P = [A_1 \quad | \quad A_2] .$$

In the above example, A_2 is an identity matrix. In general, A_2 will be lower triangular and therefore can be transformed to an identity matrix by elementary row operations.

The input to the decoder is

$$z = H' x + e ,$$

where e is the error-word pattern. Then

$$w = Hz ,$$

or

$$w = HH' x + He .$$

One requirement of the encoding and decoding scheme is to make the output of the decoder identical to the input of the encoder when $e = 0$. Or,

$$HH'x = x ,$$

$$HH' = I .$$

Now, let us assume that

$$H = \left[\begin{array}{c|c} I & O \\ \hline B_1 & B_2 \end{array} \right] ,$$

and

$$x = \begin{bmatrix} x_1 \\ 0 \end{bmatrix} ,$$

in keeping with the separable* code requirements. Then*

$$HH'x = \left[\begin{array}{c|c} I & O \\ \hline A_1 + A_2 B_1 & A_2 B_2 \end{array} \right] \begin{bmatrix} x_1 \\ 0 \end{bmatrix} = \begin{bmatrix} x_1 \\ 0 \end{bmatrix} ,$$

leaving

$$A_2 B_1 + A_1 = 0 ,$$

$$A_2 B_1 = -A_1 ,$$

$$B_1 = -[A_2^{-1} A_1] .$$

Normally A_2 is given, but since $A_2 B_2$ is indeterminate from above, we may say that B_2 is unspecified. This lack of restriction on B_2 provides more flexibility in the decoder design.

In the example we are treating, $A_2 = I$, so that $B_1 = -A_1 = A_1$. Let us choose $B_2 = I$ also. Then for this example,

$$H = H' ,$$

*A separable code is one in which the k information digits are grouped together, separate from the group of $n - k$ check digits.

* see errata sheet

where H' is given by the following matrix:

$$H' = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ \hline 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \end{bmatrix} .$$

Error Probabilities

For the single error-correcting code, we say that we have a decoding error when the number of bits in error going into the decoder is greater than 1, or when the weight* of the vector $e > 1$. By comparison, in the uncoded case, a decoding error is caused by one or more bits in error, or when the weight of $e > 0$.

For the binary symmetric channel, the probability of a bit error is p and the probability of no bit error is $1 - p$. Then, the probability of no errors in a n -bit word through the BSC (Binary Symmetric Channel) is

$$p(\text{no errors}) = (1 - p)^n ,$$

and the probability of one or more errors is

$$V_u = p(\text{1 or more errors}) = 1 - (1 - p)^n .$$

The probability of one error is

$$p(\text{1 error}) = \binom{n}{1} (1 - p)^{n-1} p ,$$

*Weight is taken here to mean the number of 'ones' in the binary vector word.

and the probability of more than one error is

$$\begin{aligned}
 & p(\text{more than 1 error}) \\
 &= p(1 \text{ or more errors}) - p(1 \text{ error}),
 \end{aligned}$$

or

$$\begin{aligned}
 V_c &= p(\text{more than 1 error}) \\
 &= 1 - (1-p)^n - np(1-p)^{n-1}.
 \end{aligned}$$

Now we can compare the coded (V_c) to uncoded (V_u) performance of the BSC for different values of p with the (13,9) code and the straight 9-bit code. Mathematically, the probability of incorrect decoding becomes unity as p approaches unity for both the coded and uncoded case. However, if we have a priori knowledge of p , and if p is in the range $0.5 < p < 1$, we can change the decoding scheme so that a 0 is decoded as a 1 and a 1 as a 0. With this scheme, the probability of incorrect decoding decreases as $p \rightarrow 1$. Both the mathematical and modified decoding error probability expressions are plotted in Figure A1.

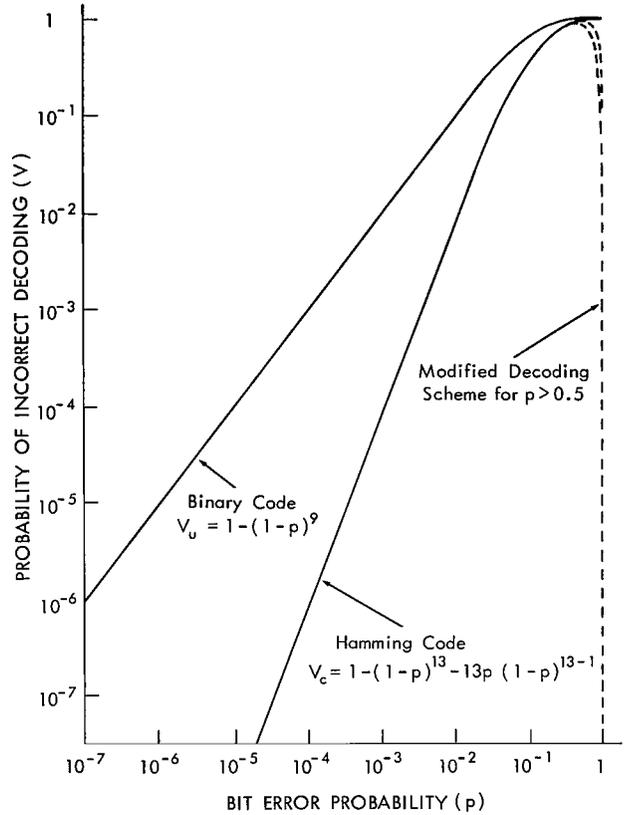


Figure A1—Comparison of a 9-bit binary code to a (13,9) Hamming code over a binary symmetric channel.

Appendix B

Example of the Minimization of a Rate Distortion Function

In his paper,* Shannon outlined a method for finding the rate distortion function $R(D)$ for a uniform discrete source and a special class of distortion-measure matrices. This method is described below and two particular distortion measure matrices which are members of the above class are used to compute and plot $R(D)$ vs. D curves.

The minimization of $R(D)$ subject to the constraints: (1) a given D , and (2)

$$\sum_Z P(Z/M) = 1 ,$$

proceeds as follows:

$$R(D) = \min I(M; Z) = \min_{M, Z} \sum P(M) P(Z/M) \log \frac{P(Z/M)}{P(Z)} ,$$

$$R(D) = \min [H(M) - H(M/Z)] .$$

We assume $P(M) = 1/M_d$, and

$$\sum_M (1) = M_d = \sum_Z (1) .$$

We also assume that the rows and columns of $[D(M, Z)]$ are permutations of the same set of $D(M, Z)$ values (a total of M_d values) and that all (M, Z) pairs having the same $D(M, Z)$ value are assigned a $P(Z/M)$ equal to the average of the $P(Z/M)$'s for all of them. We are thereby assuming a symmetric channel. Call this average conditional probability P_Z . This averaging is part of the minimizing

*Shannon, C. E., "Coding Theorems for Discrete Sources with a Fidelity Criterion," In: "Information and Decision Processes," (Robert E. Machol ed.) pp. 93-126, New York: McGraw-Hill, 1960.

process on $R(D)$ since it increases* $H(M/Z)$. Under these conditions,

$$H(M/Z) = - \sum_{M, Z} P(M, Z) \log P(M/Z) .$$

But

$$P(M/Z) = \frac{P(Z/M) P(M)}{P(Z)} .$$

Also, from the assumed probability assignment,

$$P(Z) = \sum_M P(Z/M) P(M) = \frac{1}{M_d} \sum_M P(Z/M) .$$

But

$$\sum_M P(Z/M) = 1 \text{ from } \sum_{M, Z} P(M, Z) = 1 \text{ and symmetry of } [P(Z/M)] .$$

Then

$$P(Z) = \frac{1}{M_d} = P(M) ,$$

$$P(M/Z) = P(Z/M) = P_Z ,$$

And

$$R(D) = \min \left[\log M_d + \sum_Z P_Z \log P_Z \right] .$$

Minimizing $R(D)$ by the method of Lagrangian multipliers, we set up the expression

$$U = R(D) + \rho D + \mu \sum_Z P(Z/M) .$$

*Shannon, C. E., "Coding Theorems for Discrete Sources with a Fidelity Criterion," In: "Information and Decision Processes," (Robert E. Machol, ed.) pp. 93-126, New York: McGraw-Hill, 1960.

By the above relations:

$$U = \log M_d + \sum_Z P_Z \log P_Z + \rho \sum_Z P_Z D(Z) + \mu \sum_Z P_Z ,$$

where $D(M, Z)$ has been replaced by $D(Z)$ since the assumed transitional probability matrix arrangement defines a $D(M, Z)$ for each P_Z . For $R(D)$ to reach a minimum, $\partial U / \partial P_Z = 0$, and differentiating each term in U separately (letting $\log_2 e = h$):

$$\frac{\partial}{\partial P_Z} (\log M_d) = 0 ,$$

$$\frac{\partial}{\partial P_Z} \left(\sum_Z P_Z \log P_Z \right) = \sum_Z \frac{\partial}{\partial P_Z} (P_Z \log P_Z) = \sum_Z (h + \log P_Z) = M_d h + \sum_Z \log P_Z ,$$

$$\frac{\partial}{\partial P_Z} \left(\rho \sum_Z P_Z D(Z) \right) = \rho \sum_Z \frac{\partial}{\partial P_Z} (P_Z D(Z)) = \rho \sum_Z D(Z) ,$$

$$\frac{\partial}{\partial P_Z} \left(\mu \sum_Z P_Z \right) = \mu \sum_Z \frac{\partial}{\partial P_Z} (P_Z) = \mu \sum_Z 1 = \mu M_d .$$

Then

$$\frac{\partial U}{\partial P_Z} = M_d h + \sum_Z \log P_Z + \rho \sum_Z D(Z) + \mu M_d = 0 ,$$

or

$$\sum_Z (h + \log P_Z) + \sum_Z \rho D(Z) + \sum_Z \mu = 0 ,$$

$$\sum_Z [(h + \log P_Z + \rho D(Z) + \mu)] = 0 .$$

We examine the solution to the above equation when all bracketed terms are zero, or:

$$(h + \log P_Z) + \rho D(Z) + \mu = 0 .$$

Solving for P_Z :

$$\log P_Z = -(h + \mu) - \rho D(Z) ,$$

$$P_Z = 2^{-(h+\mu)} 2^{-\rho D(Z)} .$$

Since μ is a variable multiplier, let

$$P_Z = F 2^{-\rho D(Z)} .$$

For the constraint

$$\sum_Z P_Z = 1 ,$$

let

$$F = \frac{1}{\sum_Z 2^{-\rho D(Z)}} .$$

Then

$$P_Z = \frac{2^{-\rho D(Z)}}{\sum_Z 2^{-\rho D(Z)}} .$$

Substitute this expression for P_Z in D and $R(D)$:

$$\begin{aligned} D &= \sum_Z P_Z D(Z) , \\ &= \frac{\sum_Z D(Z) 2^{-\rho D(Z)}}{\sum_Z 2^{-\rho D(Z)}} . \end{aligned}$$

$$\begin{aligned}
R(D) &= \log M_d + \sum_Z P_Z \log P_Z \quad , \\
&= \log M_d + \sum_Z P_Z \log P_Z + \log F - (\log F) \left(\sum_Z P_Z \right) \quad , \\
&= \log M_d + \sum_Z P_Z (\log P_Z - \log F) + \log F \quad , \\
&= \log M_d + \sum_Z P_Z (-\rho D(Z)) + \log F \quad , \\
&= \log M_d - \log \sum_Z 2^{-\rho D(Z)} - \rho \sum_Z P_Z D(Z) \quad , \\
&= \log \frac{M_d}{\sum_Z 2^{-\rho D(Z)}} - \rho D \quad .
\end{aligned}$$

For a given data source characterized by a set of equal probabilities $P(M)$, and a distortion measure matrix which is square and in which each row and column is a permutation of the same set of numbers, we have found a formula for the minimum source rate $R(D)$ (in bits/word) that will insure the average distortion to be less than or equal to some value D . In effect, for each value of D we compute a different set of minimizing transitional probabilities P_Z and then solve for $R(D)$. In such a way we can plot the curve of $R(D)$ vs. D . The parameter ρ fixes the value of D , and some limits are of interest. Since the M set is identical to the Z set: when $\rho = 0$,

$$D = D_{\max} \quad \square \quad \frac{1}{M_d} \sum_Z D(Z) \quad ; \quad R(D) = 0 \quad ;$$

and when $\rho \rightarrow \infty$,

$$D \rightarrow D_{\min} \quad ; \quad R(D) \rightarrow \log M_d \quad .$$

Consider the following distortion measure matrices for an 8-level system:

$$[D_1(M, Z)] =$$

Z \ M	0	1	2	3	4	5	6	7
0	0	1	2	3	4	3	2	1
1	1	0	1	2	3	4	3	2
2	2	1	0	1	2	3	4	3
3	3	2	1	0	1	2	3	4
4	4	3	2	1	0	1	2	3
5	3	4	3	2	1	0	1	2
6	2	3	4	3	2	1	0	1
7	1	2	3	4	3	2	1	0

$$[D_2(M, Z)] =$$

Z \ M	0	1	2	3	4	5	6	7
0	0	16/7	16/7	16/7	16/7	16/7	16/7	16/7
1	16/7	0	16/7	16/7	16/7	16/7	16/7	16/7
2	16/7	16/7	0	16/7	16/7	16/7	16/7	16/7
3	16/7	16/7	16/7	0	16/7	16/7	16/7	16/7
4	16/7	16/7	16/7	16/7	0	16/7	16/7	16/7
5	16/7	16/7	16/7	16/7	16/7	0	16/7	16/7
6	16/7	16/7	16/7	16/7	16/7	16/7	0	16/7
7	16/7	16/7	16/7	16/7	16/7	16/7	16/7	0

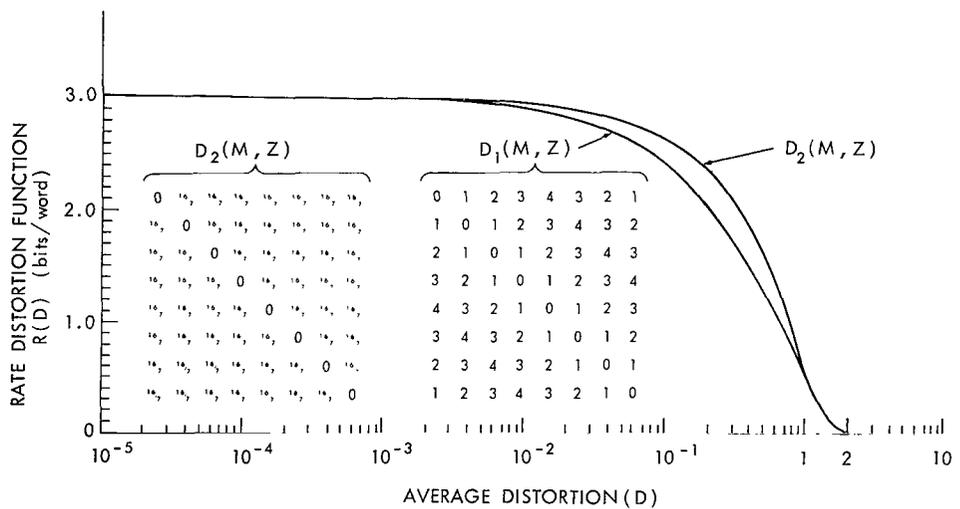


Figure B1—Rate distortion functions, $R(D)$, for a uniform probability, 8-level source with $D(M, Z)$'s as shown.

Now these matrices satisfy the above requirement of permutable row and column elements, and they both give rise to the same value of $D_{\max} = 2$. For each $D(M, Z)$ matrix we compute D and $R(D)$ for values of the dummy variable ρ and plot these rate distortion functions as shown in Figure B1.

Appendix C

Examples of Entropy Reducing Compression Effects on First-Order Probability

The following examples use two kinds of non-linear data transformations that are typical in space telemetry: limiting, and logarithmic amplification.

Example 1: Two-Level Limiter

In this case we have a very typical space data experience—that of limiting, whether intentional or unintentional. In either case, we can express two-level limiting as:

$$y = y_1 = kx_2, \quad -\infty \leq x \leq x_2.$$

$$y = kx, \quad x_2 < x < x_3.$$

$$y = y_2 = kx_3, \quad x_3 \leq x \leq \infty.$$

The effect of this non-linear transformation on the probability density function of the data, x , is shown in Figure C1. The original probability density is rectangular and can be expressed as:

$$p(x) = \frac{1}{x_4 - x_1}, \quad x_1 < x < x_4.$$

$$p(x) = 0, \quad x < x_1; \quad x > x_4.$$

As shown in Figure C1, after the limiting at levels x_2 and x_3 (where $x_2 > x_1$; $x_3 < x_4$) takes place, the limited data has a probability density function with two features: a rectangular shape within the limit levels, and an impulse shape at each limiting level.

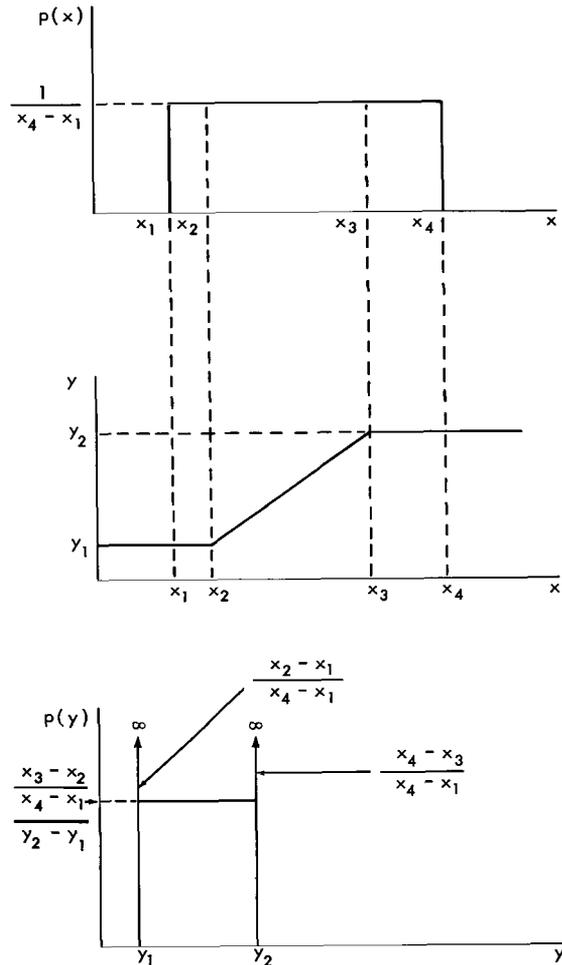


Figure C1—Entropy reducer operation: two-level limiter on a uniform probability density function.

Mathematically,

$$p(y) = \left(\frac{x_2 - x_1}{x_4 - x_1} \right) \mu_0(y - y_1), \quad y = y_1.$$

$$p(y) = \left(\frac{x_3 - x_2}{x_4 - x_1} \right) / (y_2 - y_1), \quad y_1 < y < y_2.$$

$$p(y) = \left(\frac{x_4 - x_3}{x_4 - x_1} \right) \mu_0(y - y_2), \quad y = y_2.$$

Example 2: Non-Uniform Density: Two-Level Limiter

Consider a source with probability density $p(x)$ given as follows:

$$p(x) = \frac{1}{4(x_3 - x_1)}, \quad x_1 < x < x_3.$$

$$p(x) = \frac{1}{2(x_4 - x_3)}, \quad x_3 < x < x_4.$$

$$p(x) = \frac{1}{4(x_6 - x_4)}, \quad x_4 < x < x_6.$$

and a two-level limiter, $y = f(x)$. Both $p(x)$ and $f(x)$ are shown in Figure C2.

$$y = y_1 = kx_2, \quad -\infty \leq x \leq x_2.$$

$$y = kx, \quad x_2 < x < x_5.$$

$$y = y_4 = kx_5, \quad x_5 \leq x \leq \infty.$$

The resulting probability density function of y , $p(y)$ is given by:

$$p(y) = \frac{x_2 - x_1}{4(x_3 - x_1)} \mu_0(y - y_1), \quad y = y_1.$$

$$p(y) = \left(\frac{x_3 - x_2}{4(x_3 - x_1)} \right) / (y_2 - y_1), \quad y_1 < y < y_2.$$

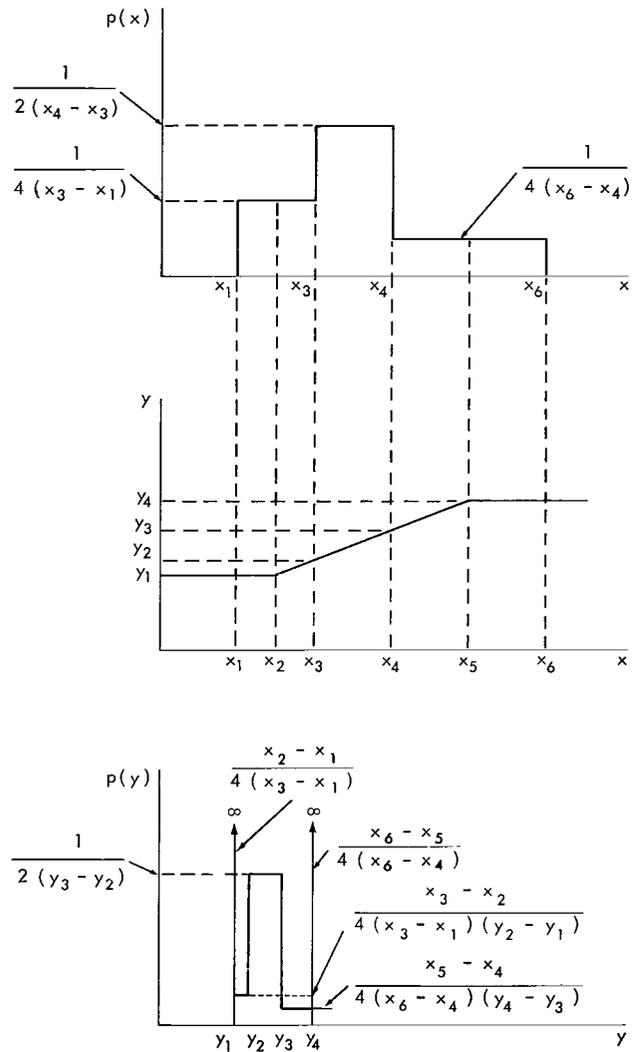


Figure C2—Entropy reducer operation: two-level limiter on a particular non-uniform probability density function.

$$p(y) = \frac{1}{2(y_3 - y_2)}, \quad y_2 < y < y_3 .$$

$$p(y) = \left(\frac{x_5 - x_4}{4(x_6 - x_4)} \right) / (y_4 - y_3), \quad y_3 < y < y_4 .$$

$$p(y) = \frac{x_6 - x_5}{4(x_6 - x_4)} \mu_0(y - y_4), \quad y = y_4 .$$

This function is also shown in Figure C2.

Example 3: Logarithmic Amplification

This transformation, although it has a mathematical inverse, qualifies as an ER compression operation since in a typical space application it reduces fidelity, and cannot be reversed to obtain the same precision as that of the original source. We may express this transformation simply as

$$y = \log x, \quad x_1 \leq x \leq x_2 .$$

$$y = 0, \quad x < x_1; \quad x > x_2 .$$

In order to determine the effect of this transformation on a uniform probability-density source, we proceed in the usual way.

The probability that x lies in $(x_0, x_0 + dx)$ must equal the probability that y lies in $(y_0, y_0 + dy)$, or

$$p(x_0) dx = p(y_0) dy .$$

$$p(y_0) = \frac{p(x_0)}{dy/dx} ,$$

or

$$p(y) = \frac{p(x)}{dy/dx} .$$

In our case

$$\frac{dy}{dx} = \frac{1}{x} = \frac{1}{e^y} ,$$

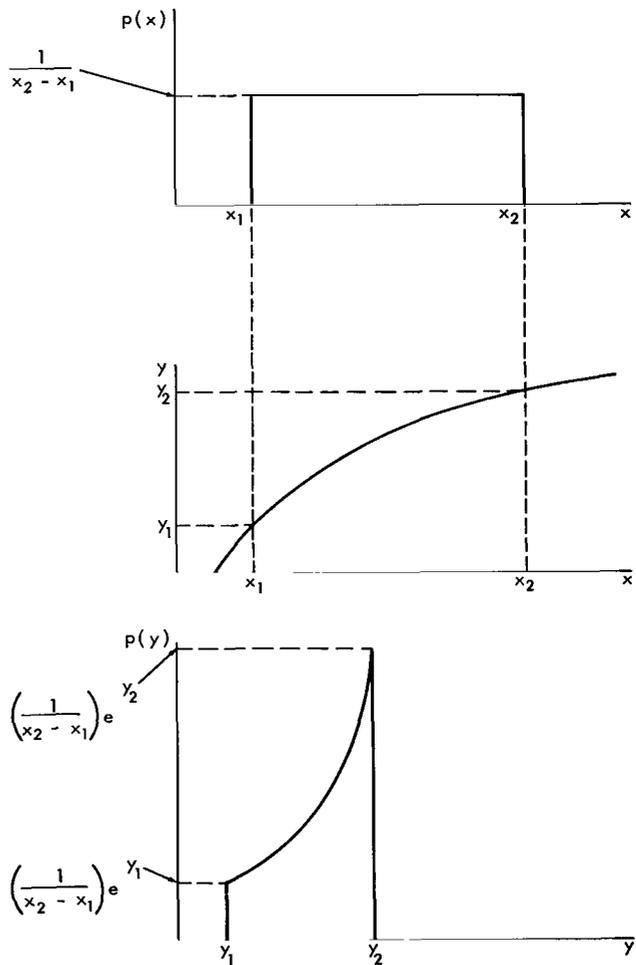


Figure C3—Entropy reducer operation: logarithmic transformation on a uniform probability density function.

then

$$p(y) = xp(x) = e^y p(x) ,$$

or

$$\left\{ \begin{array}{ll} p(y) = \left(\frac{1}{x_2 - x_1} \right) e^y , & y_1 \leq y \leq y_2 \\ p(y) = 0 , & y < y_1 ; y > y_2 \end{array} \right\} .$$

A sketch of this resulting probability density function is shown in Figure C3.

"The aeronautical and space activities of the United States shall be conducted so as to contribute . . . to the expansion of human knowledge of phenomena in the atmosphere and space. The Administration shall provide for the widest practicable and appropriate dissemination of information concerning its activities and the results thereof."

—NATIONAL AERONAUTICS AND SPACE ACT OF 1958

NASA SCIENTIFIC AND TECHNICAL PUBLICATIONS

TECHNICAL REPORTS: Scientific and technical information considered important, complete, and a lasting contribution to existing knowledge.

TECHNICAL NOTES: Information less broad in scope but nevertheless of importance as a contribution to existing knowledge.

TECHNICAL MEMORANDUMS: Information receiving limited distribution because of preliminary data, security classification, or other reasons.

CONTRACTOR REPORTS: Scientific and technical information generated under a NASA contract or grant and considered an important contribution to existing knowledge.

TECHNICAL TRANSLATIONS: Information published in a foreign language considered to merit NASA distribution in English.

SPECIAL PUBLICATIONS: Information derived from or of value to NASA activities. Publications include conference proceedings, monographs, data compilations, handbooks, sourcebooks, and special bibliographies.

TECHNOLOGY UTILIZATION PUBLICATIONS: Information on technology used by NASA that may be of particular interest in commercial and other non-aerospace applications. Publications include Tech Briefs, Technology Utilization Reports and Notes, and Technology Surveys.

Details on the availability of these publications may be obtained from:

SCIENTIFIC AND TECHNICAL INFORMATION DIVISION
NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
Washington, D.C. 20546