DATA COMPRESSION SYSTEMS

Georges E. Husson, Dip. Eng. (Elec.)

DATA COMPRESSION SYSTEMS

Electrical Georges E. Husson, Dip. Eng.(Elec.) M.Eng.

ABSTRACT

Various forms of the redundancy reduction techniques which include the zero-order and first-order predictors, are applied to digital data compression. Different schemes for supplying the timing information in a compressed system are available. In particular, the run-length encoder, single address encoder and position encoder are analyzed and compared.

It is shown that these compression systems are more sensitive to transmission errors than the conventional PCM. However, when bandwidth compression alone is considered, the performance of the compression systems can be made as good as that of PCM.

Graphs showing the effect of noise on zero-order and first-order predictors, for the three addressing schemes, are included.

DATA COMPRESSION SYSTEMS

by

Georges E. Husson, Dip. Eng.(Elec.)

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of Master of Engineering.

Department of Electrical Engineering,
McGill University,
Montreal, Quebec.

July, 1970.

ABSTRACT

Various forms of the redundancy reduction techniques which include the zero-order and first-order predictors, are applied to digital data compression. Different schemes for supplying the timing information in a compressed system are available. In particular, the run-length encoder, single address encoder and position encoder are analyzed and compared.

It is shown that these compression systems are more sensitive to transmission errors than the conventional PCM. However, when bandwidth compression alone is considered, the performance of the compression systems can be made as good as that of PCM.

Graphs showing the effect of noise on zero-order and first-order predictors, for the three addressing schemes, are included.

ACKNOWLEDGEMENTS

The author would like to express his indebtedness to Dr M.Fukada, under whose supervision this project was undertaken.

This project was supported under the National Research Council Grant # A-2735.

TABLE OF CONTENTS

CHAPTER I

DATA COMPRESSION SYSTEMS

Modern communication methods have rendered possible the transmission of analog (continuous) waveforms in digital (discrete) form. The many advantages and disadvantages offered by these techniques are now well known and have been extensively covered in the literature. Pulse-code-modulation is one way to achieve this transformation and has been applied to voice, video and telemetering data for more than a decade, thus causing a tremendous increase in the amount of digital information which must be transmitted from one point to another. The capabilities of present-day communication channel may not be sufficient to accomodate all the signals generated.

At reception, the information often appears in a format not suitable for immediate use and must, therefore, be stored before it is processed and decoded.* This results in a waste of memory space and processor's time, both costly items.

Hence, excessive bandwidth occupancy and time required for sorting the "useful" information out of this large amount of data are the two main factors that led engineers to consider more efficient ways to process digital information.

Methods were devised to remove, at least partially, the redundancy of the message by taking advantage of the degree of "predictability" which exists among the sequences

* For some expected quantities of scientific data from deep-space see reference (1).

which form the message.

Any data compression scheme that has been developed is, therefore, directly related to how the input waveform was originally sampled. We know that a bandlimited time function must be sampled at least at twice the highest frequency contained in that signal in order to extract all the information contained in the waveform. Most PCM systems are built according to that rule.

However, in the case of many telemetry signals, there aresome periods of low activity, and redundancy occurs during these periods which are grossly oversampled. Since little or no information is gained by sending the samples occurring during a low activity period, these samples are redundant and need not be sent.

Data compression or redundancy reduction consists in processing the data prior to transmission so that the received waveform can be reconstructed with a minimum number of samples to any desired accuracy. For the purpose of comparison, we will take this tolerance to lie within one quantization step of the A/D converter.

Since the occurrence of nonredundant samples is random, the data compression techniques described in the sequel usually employ temporary buffer storage of compressed data, which enables the actual transmission to be synchronous and at a rate lower than the Nyquist rate. A simplified block diagram of a data compression system is shown in Figure 1.1.

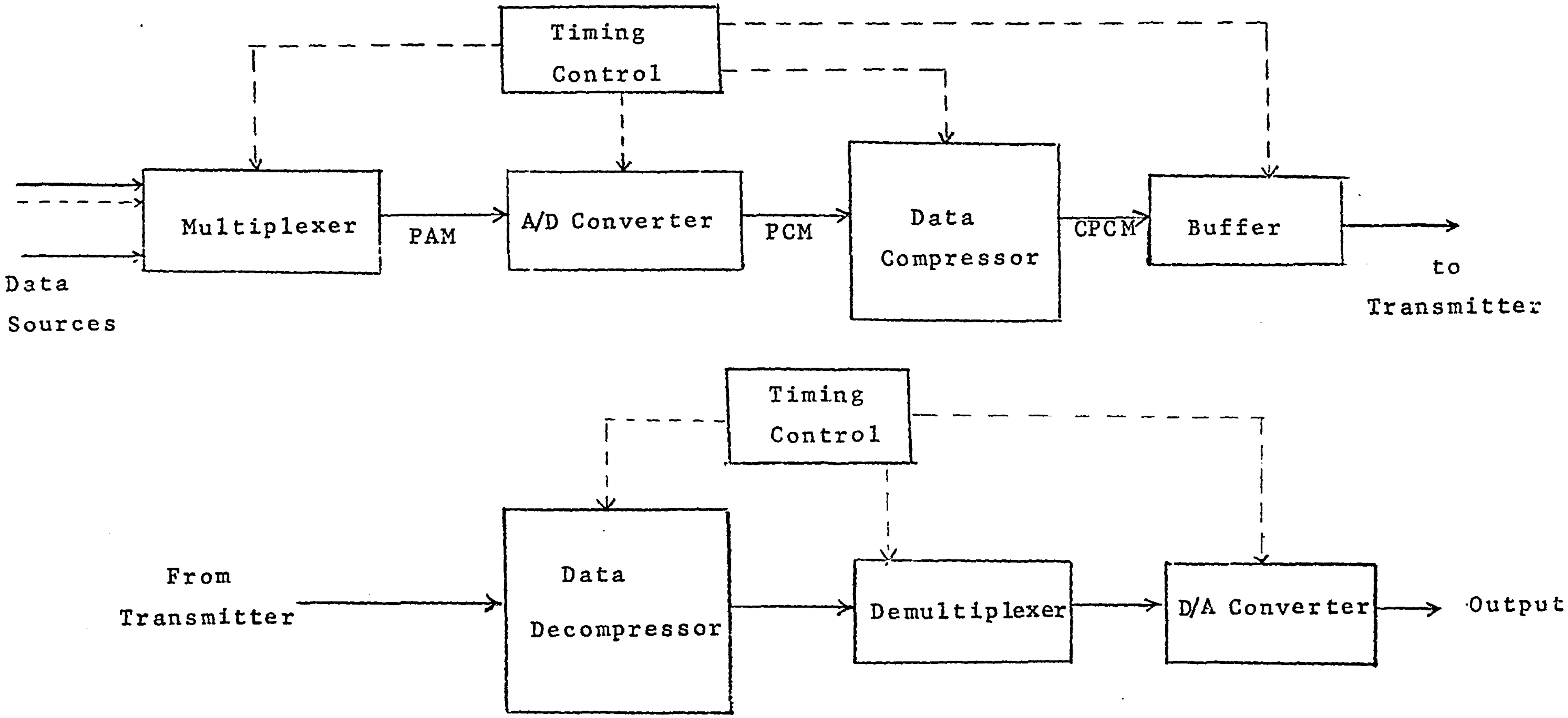


Figure 1.1 FUNCTIONAL BLOCK DIAGRAM OF A DATA COMPRESSION SYSTEM

In the course of this study we shall assume a binary symmetric channel corrupted by additive Gaussian noise.

1.1 Definitions

A data compression system will be formally defined as a communication system which adapts itself to the time-varying information content of the data and seeks to maintain an output rate which is consistent with signal activity without significantly affecting the efficiency of the transfer of information.

It appears from this definition that several systems can fultill these requirements, and hence, classification of the data compression systems is attempted in the next section. Let us first define some technical terms that will be often used in the sequel.

It was implied in the above definition that a signal can be regarded as having a varying effective bandwidth or a quasi-ststionary spectrum. A "quasi-stationary spectrum" will be defined as a short time spectrum, the mean of the signals exhibiting discontinuities between samples ("sample" is taken here in the statistical sense). It will be shown that this is a minimum requirement if adaptive methods are to be used.

To evaluate compression algorithms, we must consider some figures of merit which will serve as a basis of comparison. All things being equal, a "good" system should exhibit a high compression ratio. In the following, three compression ratios, namely, sample compression ratio, bit compression ratio and energy compression ratio, are defined, each taking care of a different aspect of data compression.

1. Sample Compression Ratio

$$(1.1) \qquad C_s = \frac{\text{Total number of samples generated}}{\text{Number of nonredundant samples transmitted}} = \frac{S_T}{S_{NR}}$$

This formula is useful to ascertain the amount of redundancy inherent in a given message, for a specific compression algorithm.

Although this figure of merit is often quoted, it does not describe the efficiency of the overall system. This efficiency can be reduced considerably if we take into account the timing information that must be sent to the receiver in order to obtain a proper reconstruction of the signal. We have already stated that we are essentially concerned in this study with synchronous systems. This entails sending the time of occurrence of each sample as well as the sample amplitude, and leads us to define a bit compression ratio.

2. Bit Compression Ratio

$$(1.2) \qquad C_B = \frac{\text{Number of bits to send (uncompressed data)}}{\text{Number of bits to send (same compressed data)}}$$

Notice that the numerator represents the number of bits sent by a fixed sampling-rate PCM system, designed to yield predetermined error fidelity criteria* Therefore, the

* The error is due mostly to the quantization noise.

ratio C_B is valid only if the waveform reconstructed from the compressed data exhibits the same quantizing noise as the PCM system, since higher bit compression ratio could be obtained if we relaxed our fidelity requirement for the compressed data and maintained a more rigid one for the non-compressed data.

The denominator of C_B consists in the number of bits necessary to represent both level and timing information. However, in (1.2), no mention has been made of synchronization bits. In digital systems, it is indeed imperative to maintain synchronization between transmitter and receiver, and this is usually achieved by sending at regular intervals a special word called the synchronization code-word. The receiver should be able to decode this word with as little ambiguity as possible, so the code word should exhibit a pattern not often encountered in the stream of information bits. Preferably, the code word should be short and its frequency of occurrence is chosen in an optimal way with respect to parameters such as noise in the channel, speed of recovery when synchronization is lost, acceptable number of synchronization loss per unit-time. We shall call the interval between two consecutive synchronization words a line. This terminology is taken from television where a line is indeed characterized by synchronization pulses marking its beginning and ending. In the following we shall assume an error-free synchronization procedure. Hence, error propagation is limited to one line. In many practical instances, this is a reasonable assumption, because enough redundancy can be inserted in the sync word to ensure accurate decoding. It is also true that the sync word will be short compared to the length of a line. Hence, we have not included the sync word in any of the expressions for the compression ratios.

We can now express formula (1.2) in the following way

$$C_B = \frac{NS_T}{S_{NR}(N+w)} = \frac{C_s}{1 + w/N} \tag{1.3}$$

where

N = number of bits per sample level

w = number of bits required for timing information

Equation (1,3) shows explicitely that C_s is an upper bound for the compression ratio of any system. The bit compression ratio C_B will approach this bound for small w. But, w depends only on the addressing scheme devised to identify nonredundant samples; we now consider the three following addressing schemes

a) Run Length Encoding A run is defined as a series of consecutive redundant samples. The run length is then the number of redundant samples in a given run. Run length encoding consists in transmitting the levels of all nonredundant samples together with binary words expressing the number of redundant samples following each nonredundant sample. The first sample in each line is always nonredundant. If a line has L samples, then $w \leq \log_2 L$. If $w < \log_2 L$, say $w=\log_2 T$ where $T < L$, some run lengths will be truncated if they exceed T bits.

Choosing the right value for T depends essentially on the bound C_s, hence, on the source statistics.

The bit compression ratio for run length encoding is

$$(1.4) \qquad C_B = \frac{C_s}{1 + \log_2 T/N}$$

b) <u>Position Word Encoding</u> The level and the address of each nonredundant sample are transmitted. The address refers to the position of the sample in the line of data.

The bit compression ratio is (here w=P , S_T=L)

$$(1.5) \qquad C_B = \frac{C_s}{1 + P/N}$$

where $P = \log_2 L$ = number of bits per address word

c) <u>Single Address Word Encoding</u> The levels of all nonredundant samples in one line are transmitted as a block. This is followed by a single address word consisting of a number of bits equal to the total number of samples per line. This L-bit word is such that a zero in the i^{th} position indicates the redundancy of the i^{th} sample, while a one in the same position indicates a nonredundant i^{th} sample.

The bit compression ratio is then given by

$$(1.6) \qquad C_B = \frac{NL}{NS_{NR} + L} = \frac{L}{S_{NR} + L/N} = \frac{C_s}{1 + C_s/N}$$

since $C_s = \dfrac{S_T}{S_{NR}} = \dfrac{L}{S_{NR}}$.

Now we wish to select the encoding scheme to obtain the largest C_B possible. This is possible only if we can assume a value for the sample compression ratio C_s, and this assumption depends on the source statistics. If the source statistics are unknown, we must rely on intuition to guide our choice.

In Figure 1.2, we have plotted the effects of the three addressing schemes on the bit compression ratio versus the sample compression ratio.

From this plot we see that for practical values of C_s and L, the position word encoding gives the lowest bit compression ratio. For small values of C_s the single address word encoding appears to be best, but run length encoding is superior to both schemes for larger values of C_s. Because it is simple to implement, run length encoding is used more frequently than any other coding methods (in particular for digital encoding of TV signals).

3. Energy Compression Ratio

Up to now, we have considered an errorless channel. However, in the practical case of a noisy channel, the degradation incurred by the compressed data may be more significant than for corresponding non-compressed data. The cause for this increased noise sensitivity can be intuitively deducted from the fact that when compression is introduced each transmitted sample represents C_s samples in the average. Thus, the reconstruction procedure will propagate an error in the sample level word over C_s samples while with regularly sampled systems only

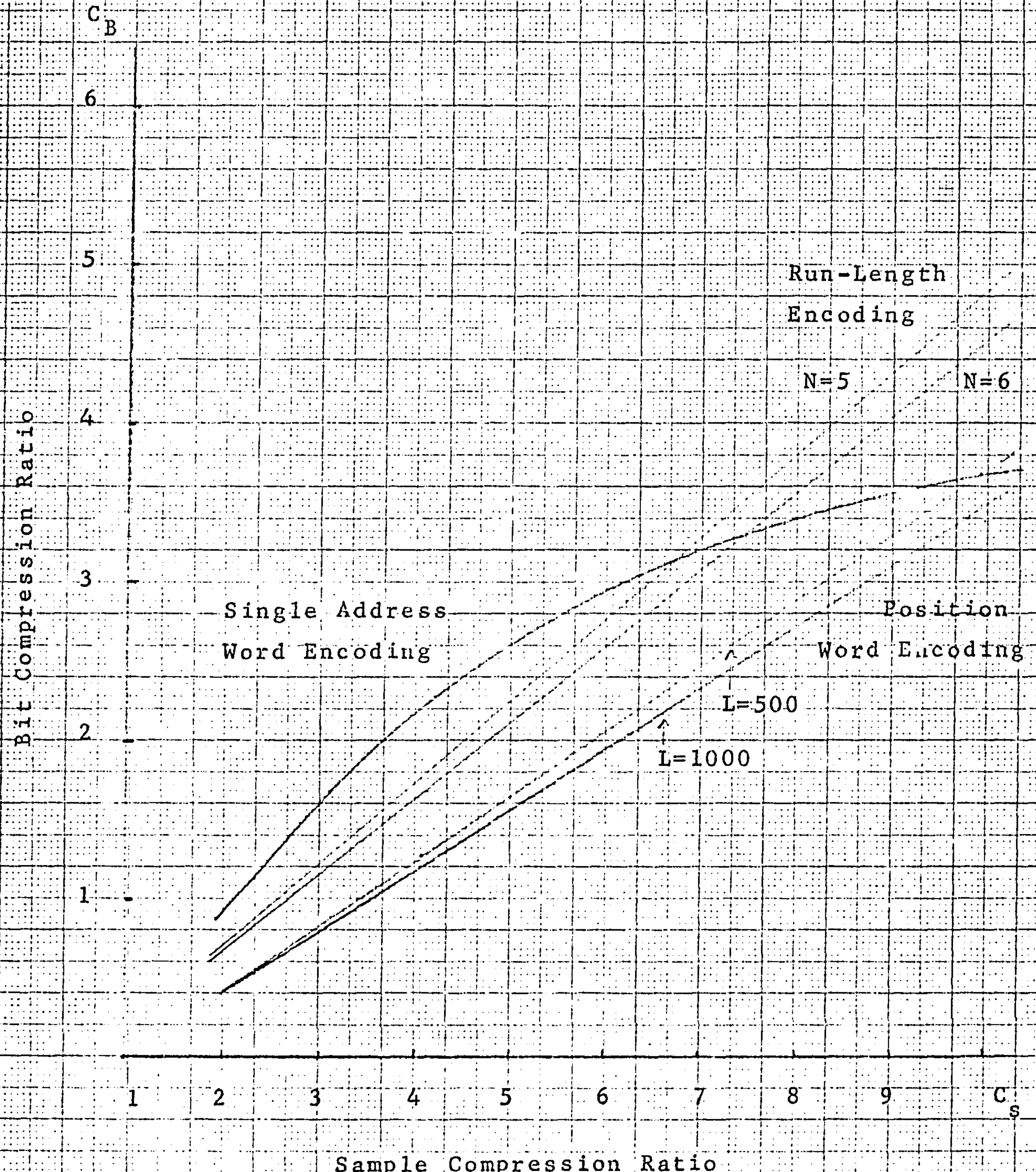


Figure 1.2 EFFECT OF ADDRESSING SCHEME ON THE BIT COMPRESSION RATIO

one sample would be affected. This is the only type of error propagation incurring in an asynchronous data compression system. It is easily evaluated and is given later. However, for synchronous data compression, there is another kind of degradation, which is due to errors occurring in the timing information that must be sent to the receiver.

We have previously assumed that synchronization is error-free, so that errors do not propagate beyond a line of data. But, an error in a run length word causes a shift in data location within the line where it occurs (this is true for any addressing scheme used). The variance of this location error increases linearly with distance from the synchronization code word. It is much more difficult to assert the influence of this type of error on the reconstructed waveform, and often one must resort to subjective tests rather than the conventional mean squared error criteria. For example, timing errors can be disastrous for compressed TV signals where entire lines are destroyed.

There are two ways to remedy this situation. First an increase in transmitter power will improve the signal-to-noise ratio and result in fewer channel errors. But, the bit compression ratio defined above does not take into account this extra energy in the signal. Davisson (2) has proposed a figure of merit called the energy compression ratio. This ratio, C_e, is defined as the ratio of the average energy required to send a sample in a non-compressed communication system to that required in a compression system for the same data quality at the receiver, and under the same noise conditions and transmission scheme. Data quality could be given in terms of r.m.s. error or probability of sample error.

The energy compression ratio is often difficult to compute; Davisson has attempted an analysis of a first-order Markov source.

Now it may happen that the transmitter power is fixed, as is often the case in telemetering applications, and the preceding trade-off cannot be accomplished. Since we are concerned with digital transmission, an alternate solution is to use error correcting codes. The insertion of some "organized" redundancy back in the compressed data could hopefully improve the signal-to-noise ratio, at the cost of decreasing the bit compression ratio. The effect of coding on C_B can be expressed as follows

$$(1.7) \qquad C_B = \frac{NL}{S_{NR}(N+w+R)} = \frac{C_S}{1 + (w+R)/N}$$

where R = number of bits allowed for coding.

1.2 Classification

Realizable data compressors fall into two main categories, namely, Entropy Reducing (ER) and Information Preserving (IP) transformations.

1. Entropy Reducing Transformations

This type of transformation performs an irreversible operation which results in an "acceptable" deterioration with respect to the fidelity criteria.

Examples of entropy reducing data compressors are narrow-band filters, limitors, vocoders (compression of speech), TV picture compressors. Generally, a special ER device must be designed for each application and no interchange is possible. An ER compressor usually operates directly on the data source, before sampling and quantization.

By definition, ER transformations reduce the fidelity of the source. But, to achieve data compression, they must also reduce the entropy of the input signal. To show that this is always true, we represent the analog source at the input of the ER device as a discrete source $X = x_i$ with M levels. This representation is valid since the thermal noise of the source and the imperfections of the instrumentation needed to measure the source characteristics limit our measurement precision. Hence, we can only distinguish M states of the source (M may be quite large but is bounded).

Now, the entropy of the source is

$$H(X) = - \sum_{i=1}^{M} P(x_i) \log P(x_i)$$

and if we denote by Y the output of the ER device, we may write

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

Noting that $y_i = f(x_i)$, we have $H(Y/X) = 0$. But, since

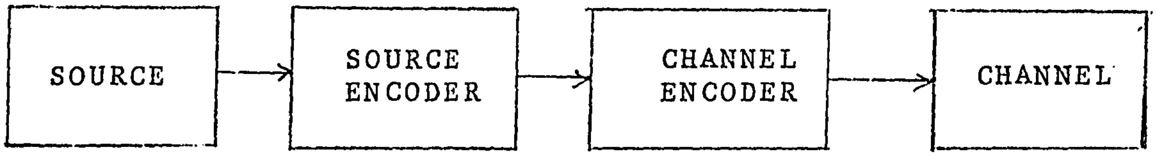
ER transformations are irreversible, $H(X/Y) > 0$. It follows that

$$H(Y) = H(X) - H(X/Y) < H(X)$$

which proves that ER compression results in a reduction in entropy.

A narrow-band low-pass filter is often used to achieve ER compression. Indeed, one can show that the maximum entropy of a source is proportional to the dimensionality of the signal space and a filter reduces this dimensionality.

Before passing to IP compression methods, we briefly describe some source encoding tcchniques. This will lead us to define an ideal compression ratio. We divide the coding of information into two parts, as shown below



Channel encoding consists in inserting some controlled redundancy into the information flow so as to combat noise more efficiently.

Source encoding directly influences the bit rate of the transmission through the channel. Assume a band-limited white Gaussian process, which is sampled at the Nyquist rate and quantized optimally to M levels.

(i) Binary Encoding - The M quantization levels are encoded into r binary digits where $M \leq 2^r$.

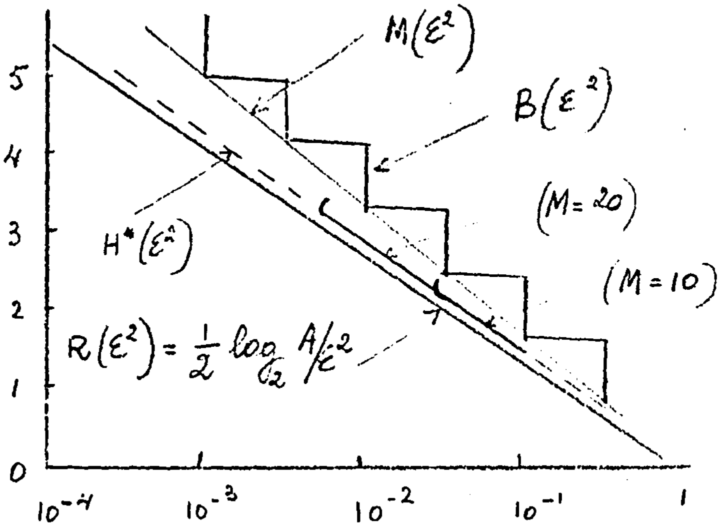
(ii) M-ary Encoding - Rather than encoding each sample separately, a block of k samples is encoded at once, where k is such that $M^k = 2^h$ for some h.

(iii) Entropy Encoding - The quantization levels do not have the same probability. We take this into account by entropy encoding; it consists in what Oliver (3) has coined as N-gramming. If a quantization level has probability of occurrence p_i, we assign log p_i binary digits to its encoding. Hence, shorter codes are assigned to the more probable levels. The Shannon-Fano code and the Huffman code are typical examples of N-gramming.

Now, for each of these schemes, there is an optimum number of quantization levels M for a given mean square error, i.e., a value of M that will minimize the rate at which it is necessary to transmit information. Goblick (4) has analyzed these schemes and arrived at some curves which are reproduced in Figure 1.3.

It is known that Shannon's rate distortion function yields the minimum possible transmission rate for a given data error. The rate distortion lower bound $R(\epsilon)$ is also plotted in Figure 1.3, thus showing that entropy encoding requires only .25 bits per sample more than the lower bound.

However, the efficiency of entropy encoding depends strongly on the source statistics, and hence, it can only be applied safely to highly correlated stationary sources. A time-varying signal could result in certain cases in a band-



SOURCE $p(u) = \exp\left(-u^2/2A\right)/\left(2\pi A\right)^{1/2}$

$$H(V) = -\sum_{1}^{M} p_i \log_2 p_i$$

$$H^*(\varepsilon^2) = \frac{1}{4} + \frac{1}{2}\log_2 \frac{A}{\varepsilon^2} \text{ bits/sample}$$

Figure 1.3 DATA RATE FOR BINARY CODING

width expansion. To achieve an efficient coding, the designer must then resort to adaptive methods, and a sub-optimal system is usually obtained. Adaptive coding is a sub-optimal procedure which consists in monitoring continuously the source statistics and updating the coding procedure according to these measurements. Hence, the past history of the signal is used in the determination of future code assignments. The decision rule which performs this mapping need not be adaptive itself and a fixed rule known to both the transmitter and the receiver greatly facilitates the design of the system, since it is then unnecessary for the transmitter to send information on how and when this rule has varied. However, the source must be quasi-stationary if we expect a limited number of measurements to converge to some useful statistics.

This procedure results, therefore, in a compromise between the optimum value set by the rate distortion function (attainable when the signal statistics are completely known) and the maximum entropy coding which transmits the total information (straight PCM).

In the next section, a practical implementation of entropy encoding for picturial data is briefly described. We simply observe here that the measurement of signal statistics can approach ideal coding for ergodic processes (like TV signals). In the more practical case of quasi-stationary processes, one should detect only the fast transients of the data, since experiments have shown that coding assignments are not influenced by neighbouring statistics (5).

Although the complexity of entropy encoding has led designers to prefer other methods of data compression, the concept has proved useful in establishing some bounds with respects to compression ratios. In particular, it is possible to define an ideal compression ratio* which does not depend on the procedure used to perform the data compression. The ideal compression ratio is defined as the maximum source information rate in the absence of any compression algorithm, divided by the entropy of the source. Now it is well known that the maximum entropy of a source $X = \{x_i\}$, where $i = 1, 2, \ldots, M$, is obtained when all symbols are equally probable, or equivalently, when there is no redundancy in the signal. In that case $P_i = 1/M$ and

$$H_{max} = -\frac{1}{M} \sum_{i=1}^{M} \log_2 M = \log_2 M$$

But, the actual entropy of the source is

$$H = -\sum_{i=1}^{M} P_i \log_2 P_i$$

Hence, the ideal compression ratio can be expressed as

$$C_{ideal} = \frac{\log_2 M}{-\sum_{i=1}^{M} P_i \log_2 P_i}$$

* sometimes called "optimum compression ratio"

When the source statistics are known, the ideal compression ratio can be calculated and the performance of various data compression algorithms can be compared to C_{ideal} which is an upper bound for all possible algorithms (note that the sample compression ratio C_s is an upper bound only for the bit compression ratio achieved by a given procedure).

Finally, we note that entropy coding, as well as the information preserving transformations which will be defined shortly, causes an increase in entropy. Indeed, letting R denote the redundancy of the source, we have

$$R_X = 1 - H_X/\log M$$

and after compression

$$R_Y = 1 - H_Y/\log M$$

But, we must have $R_Y \geqslant R_X$; thus $H_Y \geqslant H_X$. This is due to the fact that adjacent samples in the compressed data are less correlated than before compression when prediction has been successful.

The ideal compression ratio can be expressed in terms of the source redundancy R , in the following way

$$C_{ideal} = \frac{1}{1 - R_X}$$

For more information on this subject, the reader is referred to the literature (6) (7).

2. Information-Preserving Transformations

Information-preserving (IP) transformations are a reversible mapping of a set of message symbols into a set of sequences containing less binary digits. The signal can always be reconstructed exactly and the choice of the coding procedure, if redundancy is to be removed, depends essentially on the signal statistics. However, the exact nature of this dependency is not usually known and there is no unique solution for an optimum mapping. Often, a method of trial and error will result in the desired procedure, but for certain input waveforms (e.g., voice and TV signals) a mathematical model is found very useful.

Since IP data compression reduces the number of samples that must be transmitted, it also reduces the energy required to transmit the source information within some tolerated error criteria.

The basic types of compression exhibiting these features are polynomial curve fitters, statistical predictors and adaptive samplers (8-13).

A. Polynomial Curve Fitting

(i) Polynomial Predictors This method involves the approximation of the signal between sample points by a polynomial and is mathematically equivalent to an interpolation process. If we let X_t represent the predicted value of a sample at time t, we can write the following difference equation

$$(1.8) \quad \hat{X}_t = X_{t-1} + \Delta X_{t-1} + \Delta^2 X_{t-1} + \ldots + \Delta^n X_{t-1}$$

where X_{t-1} is the value of the sample occurring at time t-1,

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

$$\Delta^2 X_{t-1} = \Delta X_{t-1} - \Delta X_{t-2}$$

$$\cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot$$

$$X_{t-2} = X_{t-2} - X_{t-3}$$

$$\cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot$$

$$\Delta^n X_{t-1} = \Delta^{n-1} X_{t-1} - \Delta^{n-2} X_{t-2} \quad .$$

Equation (1.8) implies that X_t is predicted according to the value of the (n+1) previous samples.

The simplest form of predictor is the zero-order predictor (ZOP) given by (n=0)

$$\hat{X}_t = X_{t-1}$$

It represents the largest possible set of consecutive data samples within an accepted error tolerance, by a horizontal straight line. In practice, a tolerance band or "aperture" K is placed about the preceding sample. This aperture is usually taken equal to or a multiple of the quantization step. If the level of the sample at time t exceeds the level of the previous sample by an amount equal to the aperture K, then it is judged nonredundant and is transmitted. This nonredundant sample forms now the new reference for comparing the ensuing samples. Otherwise, it is discarded as redundant, and hence, is not transmitted. The algorithm for the zero-order predictor is illustrated in Figure 1.4.

Note that according to this algorithm, timing information must be sent with each nonredundant samples. For the signal shown above there are eight nonredundant samples, hence eight timing words must be transmitted along with the eight level words. It is possible, however, to modify the preceding algorithm so that fewer timing words will be required for the reconstruction of certain types of data. The modified algorithm is shown in Figure 1.5.

It is easy to show that in no case will the modified algorithm require more timing words than the preceding algorithm. For the portion of signal given in Figure 1.5, two timing words are transmitted (instead of eight). However, for the modified algorithm, a flag is required to differentiate level information from timing information and this could increase the length of the timing and level words by one bit.

Another form of polynomial predictor is the first-order predictor. In this case, we have

$$\hat{X}_t = X_{t-1} + \Delta X_{t-1}$$

where

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

There are several methods for representing redundant samples by a straight line segment (15). We give here only one method, as illustrated in Figure 1.6 . The mechanization of this pro-

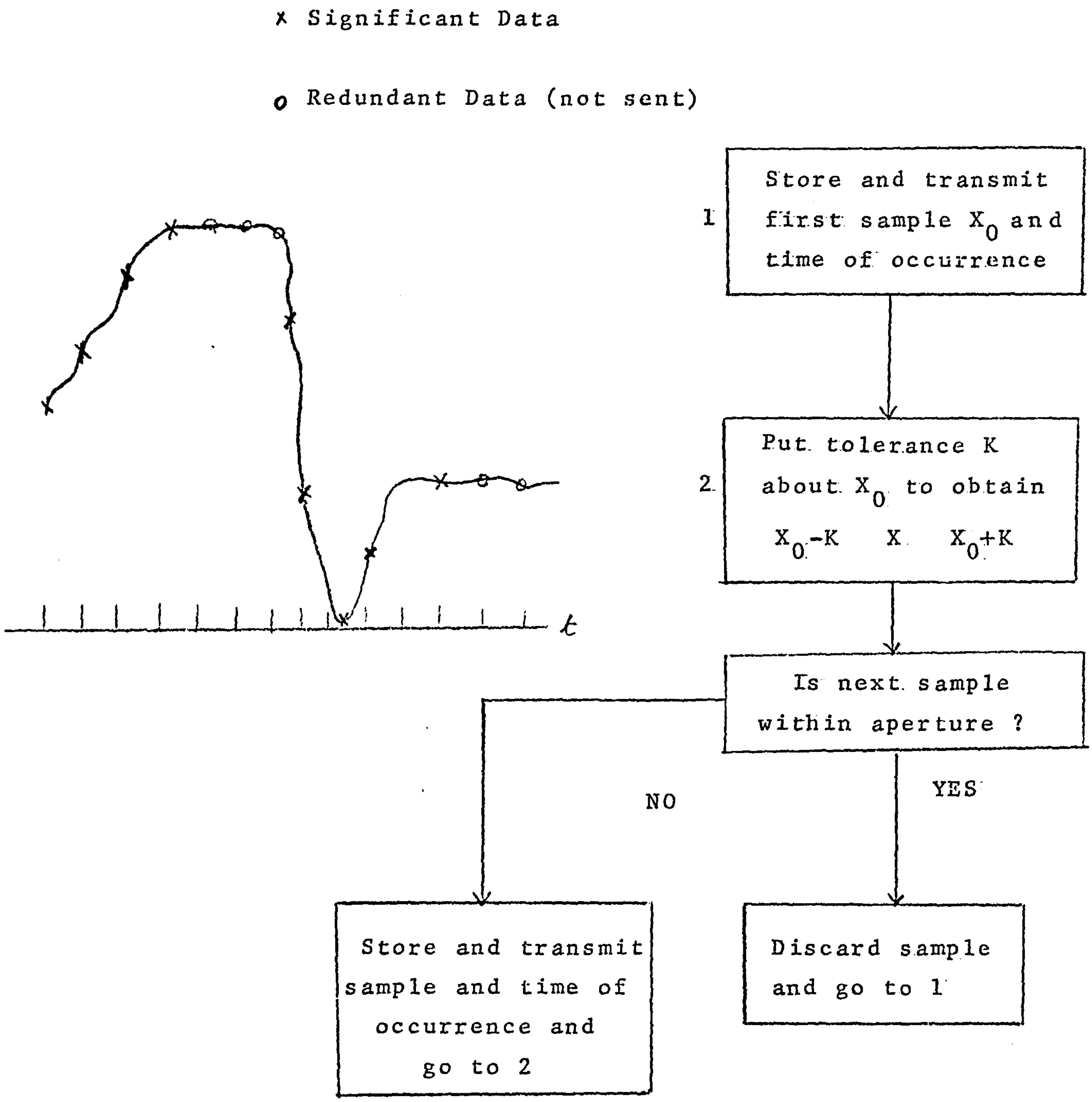

x Significant Data
o Redundant Data (not sent)
t
1
Store and transmit first sample X_0 and time of occurrence
2
Put tolerance K about X_0 to obtain
X_0-K X X_0+K
Is next sample within aperture ?
NO
YES
Store and transmit sample and time of occurrence and go to 2
Discard sample and go to 1

Figure 1.4 ZERO - ORDER PREDICTION ALGORITHM

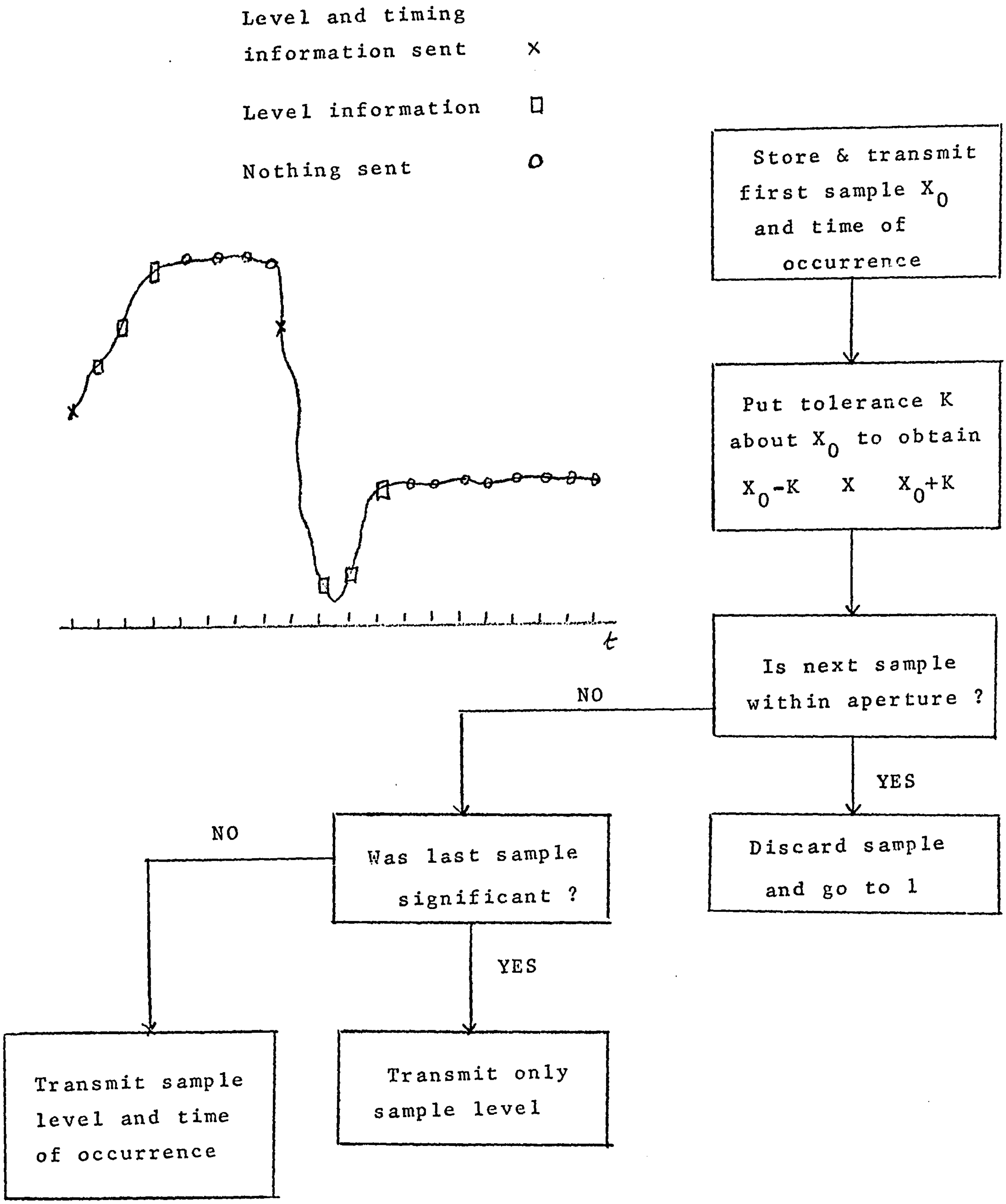

Level and timing information sent ×
Level information □
Nothing sent o
t
Store & transmit first sample X_0 and time of occurrence
Put tolerance K about X_0 to obtain
X_0-K X X_0+K
Is next sample within aperture ?
NO
YES
Discard sample and go to 1
Was last sample significant ?
NO
YES
Transmit sample level and time of occurrence
Transmit only sample level

Figure 1.5 MODIFIED ZERO-ORDER PREDICTOR

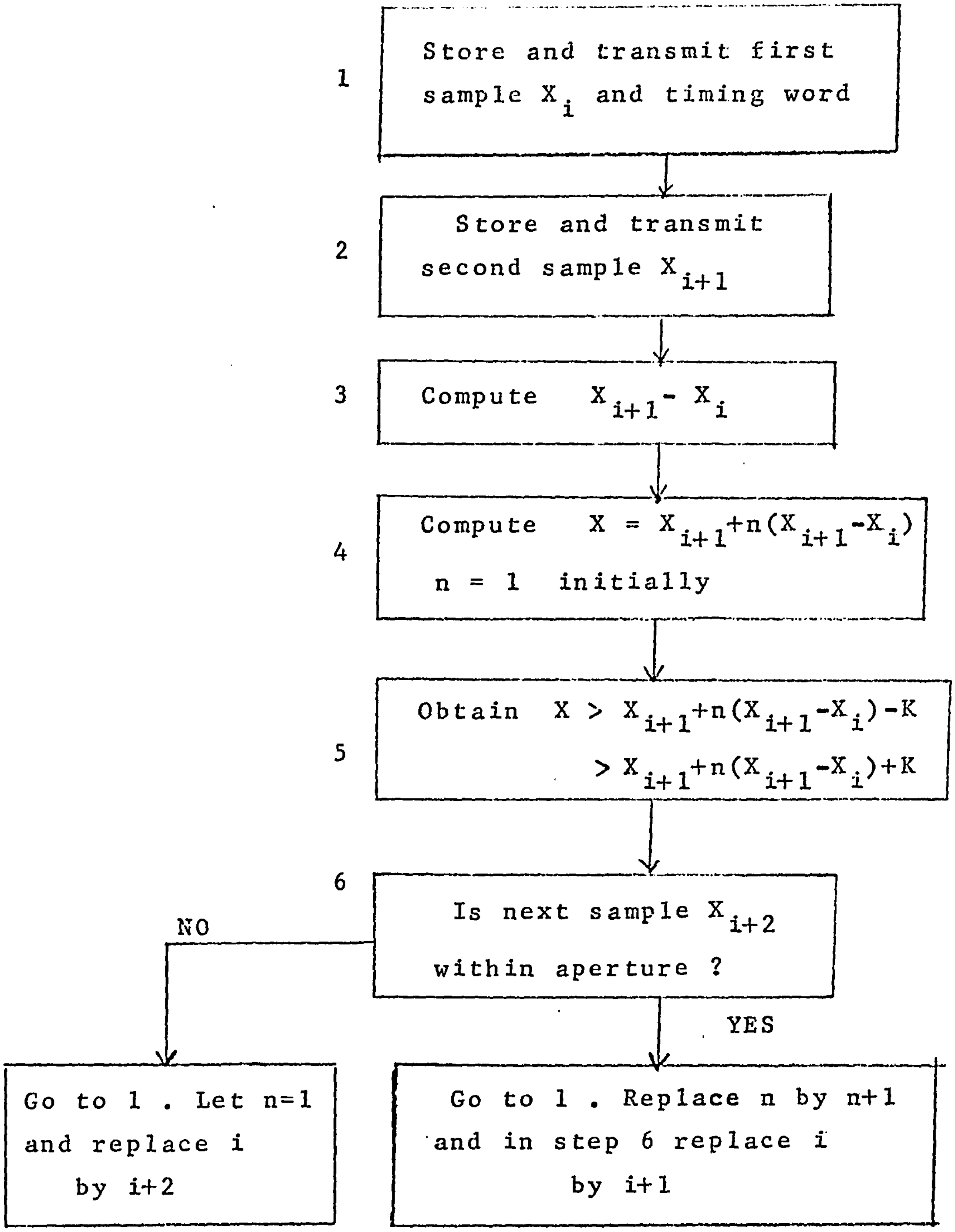

1
Store and transmit first sample X_i and timing word
2
Store and transmit second sample X_{i+1}
3
Compute $X_{i+1} - X_i$
4
Compute $X = X_{i+1} + n(X_{i+1} - X_i)$
n = 1 initially
5
Obtain $X > X_{i+1} + n(X_{i+1} - X_i) - K$
$> X_{i+1} + n(X_{i+1} - X_i) + K$
6
Is next sample X_{i+2} within aperture ?
NO
YES
Go to 1 . Let n=1 and replace i by i+2
Go to 1 . Replace n by n+1 and in step 6 replace i by i+1

Figure 1.6 FIRST-ORDER PREDICTOR ALGORITHM

cess will be given in the next section.

(ii) Polynomial Interpolator The difference between interpolator and predictor is that for interpolators the interpolation is affected by the sample values between the last transmitted value and the present one.

Zero-Order Polynomial Interpolator As for the zero-order predictor, the redundant portions of the input signal are represented by a straight line, but the difference exists in the choice of the reference sample to represent the redundant set. The reference sample for the interpolator is determined at the end of the redundant set, whereas for the predictor it was the first sample. Also, the reference sample X_t in the interpolator is the average between the largest sample X_l and the smallest X_s in the redundant set.

B. Statistical Predictors

Statistical predictors involve prediction of samples by weighting a set of previous samples in some specified manner. The difference between this method and the types previously mentionned is that the prediction rule is self-controlled. It is determined by a learning process which makes use of a set of previous samples not necessarily immediately prior to the predicted sample. An optimum predictor is, in principle, possible if the power spectrum of the signal is known exactly. If not, one can expect a sub-optimum predictor.

The prediction equation could be represented by some linear combinations of past samples

$$X_t = \sum_{k=1}^{M} a_k X_{t-k} .$$

This is the equation of a linear nonrecursive filter and the coefficients could be obtained by solving the Wiener-Hopf equation for discrete data.

More details on statistical predictors are to be found in (14) and (15). This method is too complex for practical hardware implementation and the results have been obtained by computer simulation.

C. Adaptive Samplers

One way to eliminate redundancy in a sequence of samples is to adjust the sampling rate to the information content of the source, since this would produce nonredundant samples only. But this demands complete knowledge of the source statistics and generally telemetry systems are greatly oversampled. Another drawback of this method is that after each change in the sampling rate, there exists a period during which the signal cannot be accurately reconstructed. This procedure, though theoretically interesting, has, therefore, not been implemented.

Before concluding this section, we shall add to the preceding classification some methods which can, at least in principle lead to data compression.

1. Transformation Compressors A transformation is performed on the analog or digital data by nonlinear or linear transformation. At the receiver, decompression is obtained by applying an inverse transformation. Logarithmic amplifiers, filters and compounders are practical examples of transformation compressors. Other types of transformation compressors are Fourier filtering and Karhuncn-Loeve compressors. Because of their complexity, the last two methods have not been implemented.

2. Parameter Extraction Compressors The method consists in extracting a particular parameter from the signal and transmitting this parameter alone. This process is irreversible since the original data cannot be reconstructed from the transmitted parameter.

3. Bit-Plane Encoding (16) This interesting method consists in partitioning the information bits into subgroups so that some of the subgroups can be encoded efficiently. We shall see in the next section how this method is implemented. The method has proved useful when the amplitude spectrum of the data is concentrated in different ranges in different time intervals. It is an information preserving method.

1.3 Implementation

We consider in this section the implementation of some data compression systems based on redundancy reduction and entropy reduction techniques.

1. Redundancy Reduction A simplified block diagram common to all redundancy reduction schemes is shown in Figure 1.7. The reference memory stores all data which will serve to perform the compression; they are previous samples, tolerance limits, slope limits, selection of a particular algorithm, etc. The comparator determines then whether each new sample is redundant or nonredundant, and updates the reference memory accordingly. The nonredundant values are sent to the buffer memory, which permits synchronous transmission through the channel. The design of the output buffer is an important and often difficult task. The size of the buffer is proportional to the expected overall compression efficiency of the system and to the maximum degradation acceptable in the reconstructed waveform. Indeed, even for stationary sources, the observation over a short interval of the stream of redundant samples often indicates a large deviation from the average flow. This will cause either overflow or underflow of the buffer. Overflow is the most serious drawback because it causes the loss of nonredundant samples, and since the redundancy of the data has already been reduced. Several studies exist on the subject of optimum and adaptive buffering. References (17) and (18) present an excellent coverage of the various techniques available.

The block marked as "timing and control" provides the necessary signals to control the sequence of operations which the data compression system must perform. The timing signals are derived through logic circuitry from a clock.

Figure 1.8 shows a more elaborate block diagram of a typical telemetering data compression system, while Figure 1.9 is a detailed description of the data compressor.

In Figure 1.8, the block marked as "queue monitor" delivers a control signal which is a function of buffer occupancy. This signal is used to adjust the aperture tolerance initially set for optimum performance in such a way as to decrease the data rate at the compressor output. This method, due to Massey (19), reduces the accuracy of the output data regardless of the data activity. Buffer overflow can, however, be controlled by adaptive filtering of the input signals.(Input signals are often filtered prior to multiplexing to avoid aliasing errors due to sampling) Adaptive filtering would cause additional degradation only in the high activity part of the data, which would otherwise cause the buffer to overflow.

2. Adaptive Methods

Adaptive Predictor

The general block diagram of an adaptive predictor compression system is given in Figure 1.10; this technique was first suggested by Balakrishnan and applied to picturial data via computer simulation (20). The system is essentially an adaptive ER transformation, in this case a predictor, which consists in an arithmetic unit, a memory and a control signal generator. Each sample S_a is compared with its predic-

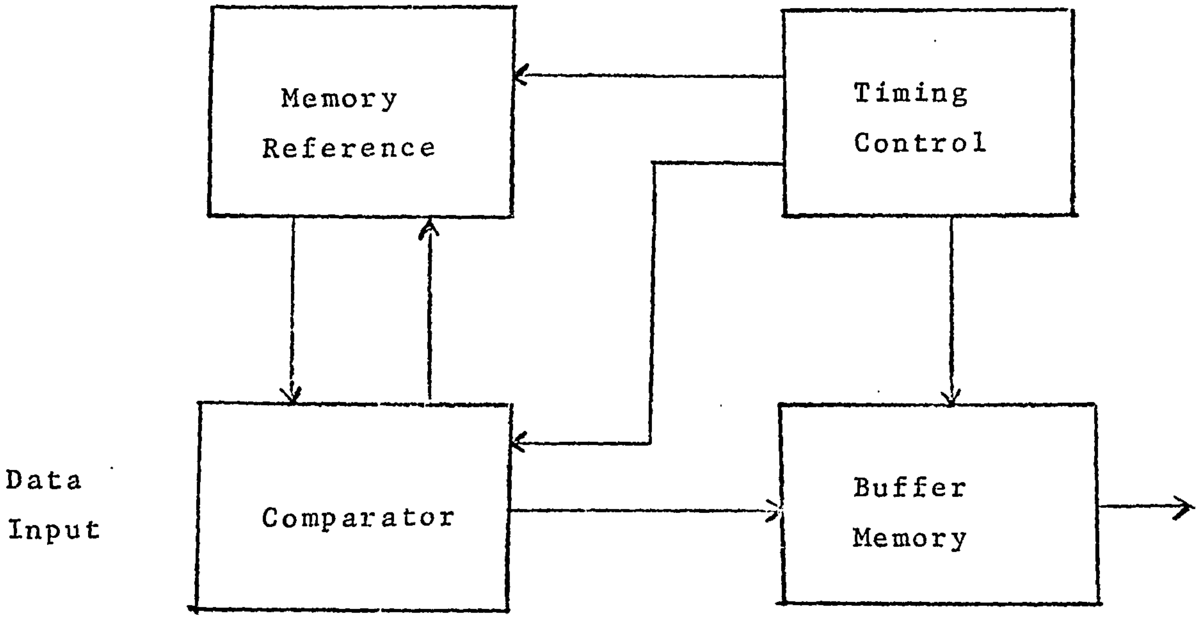


Figure 1.7 REDUNDANCY REDUCTION BLOCK DIAGRAM

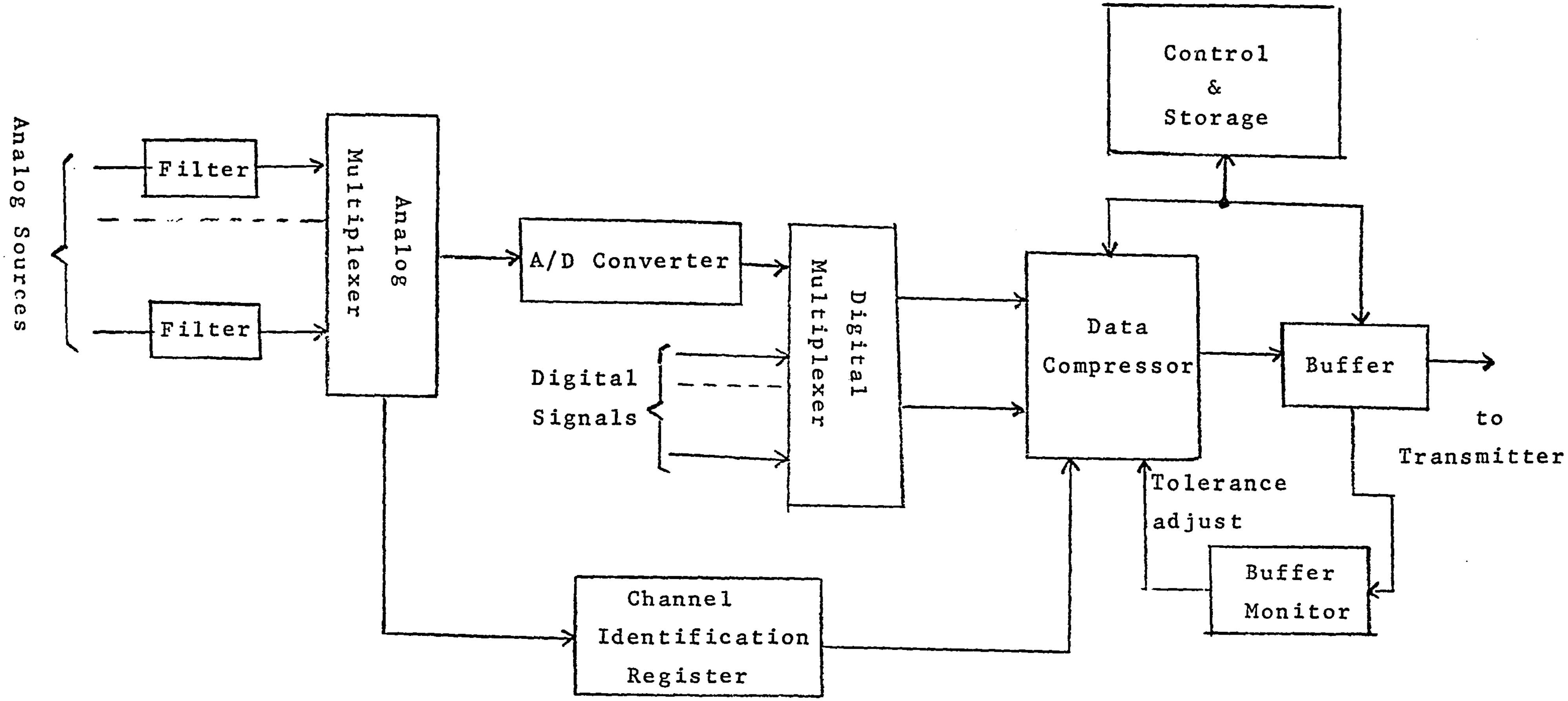


Figure i.8 BLOCK DIAGRAM OF A TELEMETRY DATA COMPRESSION SYSTEM

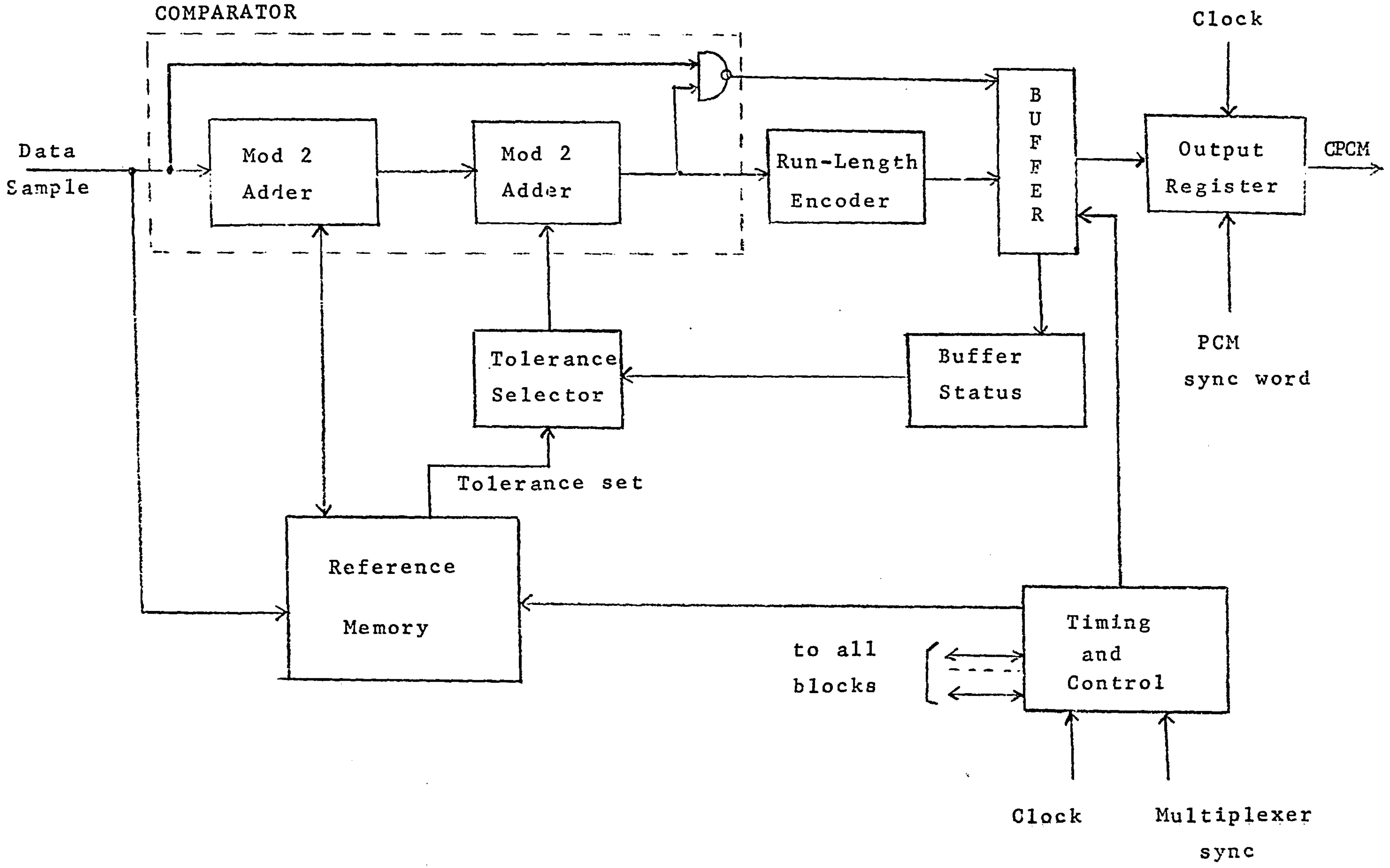


Figure 1.9 BLOCK DIAGRAM OF DATA COMPRESSOR

ted value S_p and the prediction error $e_p = S_a - S_p$ is obtained. The value of e is then compared to some predetermined error threshold, Q; if $e_p > Q$ the sample in question is significant and must be transmitted; if $e_p < Q$ the sample is predictable and hence redundant. This part of the system is similar to the preceding polynomial predictors described previously. The fundamental difference is in the feedback from the comparator output to the predictor which serves to update the prediction mechanism. The updating can be accomplished in several ways. For example, if we want to predict the k^{th} sample S_k, having observed the m preceding samples, we could try to obtain the best nonlinear estimate for S_k in the m.s. sense, given by

$$\hat{S}_k = E(S_k / S_{k_1}, S_{k_2}, \ldots, S_{k_m})$$

or

$$(1.9) \qquad \hat{S}_k = \sum_{i=1}^{m} i \Pr(S_k = i / S_{k_1}, S_{k_2}, \ldots, S_{k_m})$$

where i denotes the i^{th} quantum level.

The memory of the predictor should be updated so that the conditional probability given in (1.9) be estimated from the data. It follows that the predictor's efficiency will be proportional to the storage capability of the memory; the larger the size m of the memory, the better will be the estimate of the conditional probability, hence that of $\hat{S}_k$. In practice, since there are M^m possible observations of the vector $S_{k_1}, \ldots, S_{k_m}$, the size m of the memory is limited to three.

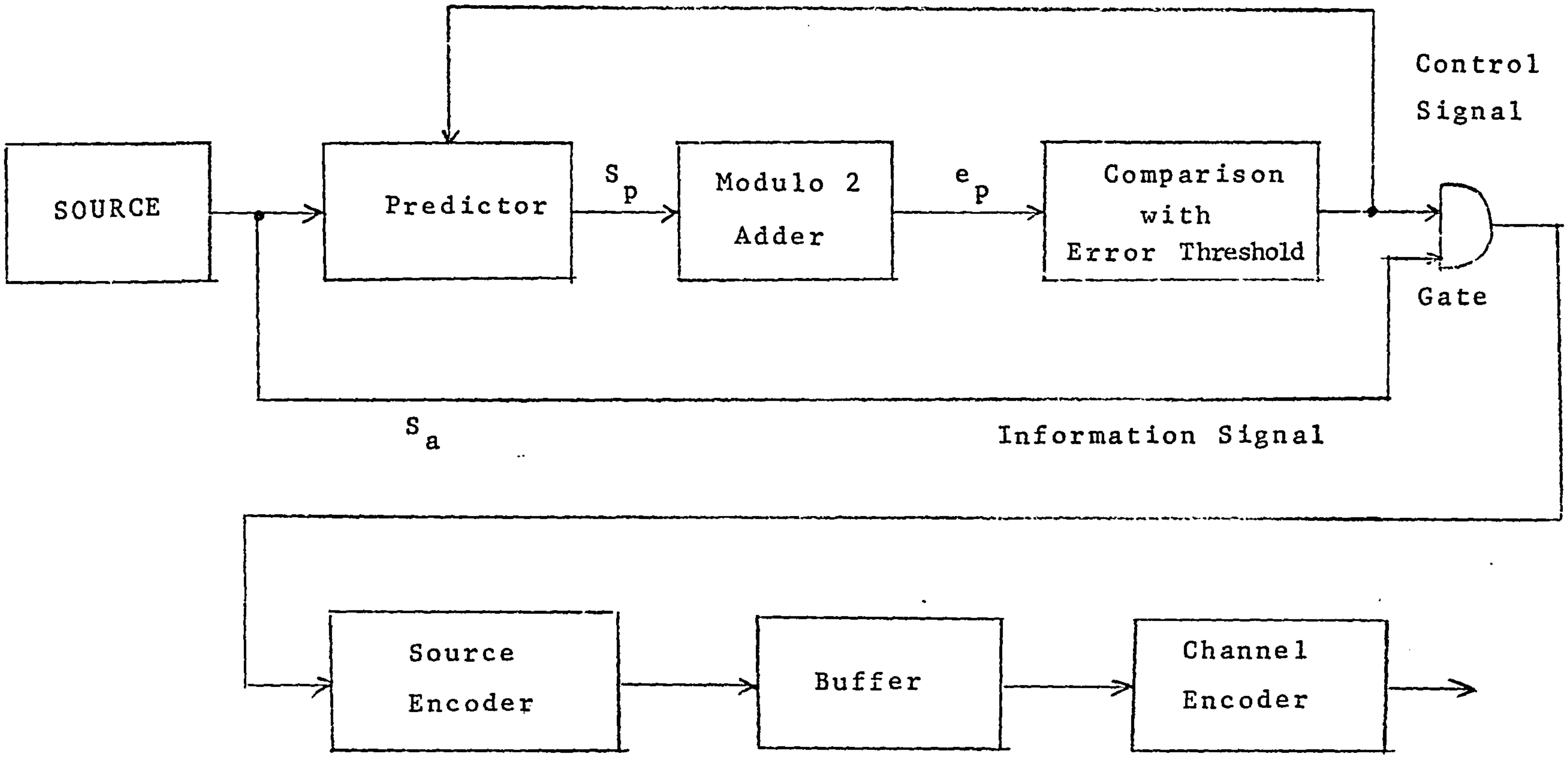


Figure 1.10 ADAPTIVE PREDICTOR

Other methods to achieve prediction are given in (21). We describe now an adaptive coding procedure which can be used in conjunction with the preceding method.

Adaptive Coding

Source encoding is this case by entropy encoding. The adaptive method consists in measuring the efficiency of the coding procedure and determining a new procedure according to the result of this measurement.

As for the previous method, an error signal must be produced and fed back to the source encoder. This error can be obtained in the following way.* Let $\{p_i\}$ be the probability of occurrence of a sequence $\{x_i\}$ and $\{q_i\}$ the probability of occurrence of $\{y_i\}$. If the sequence $\{y_i\}$ occurs and is encoded with $-\log_2 p_i$ bits per symbol, then the excess number of bits used for the i^{th} sequence is

$$\delta_i = \log_2 q_i - \log_2 p_i = \log_2 p_i/q_i \ .$$

The average number of bits per sequence in excess is

$$\Delta H = \sum_i q_i \delta_i = \sum_i q_i \log_2 q_i/p_i \ .$$

Now, after the measurement of S_T realizations, the best estimate of the q_i is given by $\lambda = S_i/S_T$, where S_i is the number of occurrences of the i^{th} sequence out of the S_T realizations. Initially, we code for maximum entropy ($p_i = 1/M$). Then the initial excess is

* This method has been suggested by Blasbalg and Van Blerkom (5).

$$\Delta H_1 = \sum_{i=1}^{M} \log_2 \lambda_i M = M \log_2 M + \sum_{i=1}^{M} \lambda_i \log_2 \lambda_i .$$

If $\Delta H_1 \geqslant \Delta H_0$, the maximum entropy coding is inefficient and we code with $p_i = \lambda_i$. If $\Delta H_1 < \Delta H_0$, we continue with the initial code.

The new measurement now yields

$$\Delta H_2 = \sum_{i=1}^{M} \lambda_i \log_2 \lambda_i / p_i$$

For samples of reasonable size, it has been shown that ΔH has a chi-square distribution of (N-1) degrees of freedom. The sensitivity of ΔH to variations of λ can be obtained from the following equation

$$\Delta H = \log_2 \lambda / p + (1-\lambda) \log_2 \frac{1-\lambda}{1-p} .$$

The plot shown in Figure 1.11 illustrates the excess bits as a function of p when the true probabilities are λ. It is seen from these curves that there is no excess when coding matches the statistics. Also, the curves are flat near the minimum, indicating that the coding is not sensitive to small deviations from the exact probabilities λ. Therefore, the statistical estimates λ_i could be obtained from a relatively small set of samples.

A functional block diagram of an adaptive coder is shown in Figure 1.12. The output of each block is explicitely stated and the overall operation follows the description given above. A control line has been added to adjust the source en-

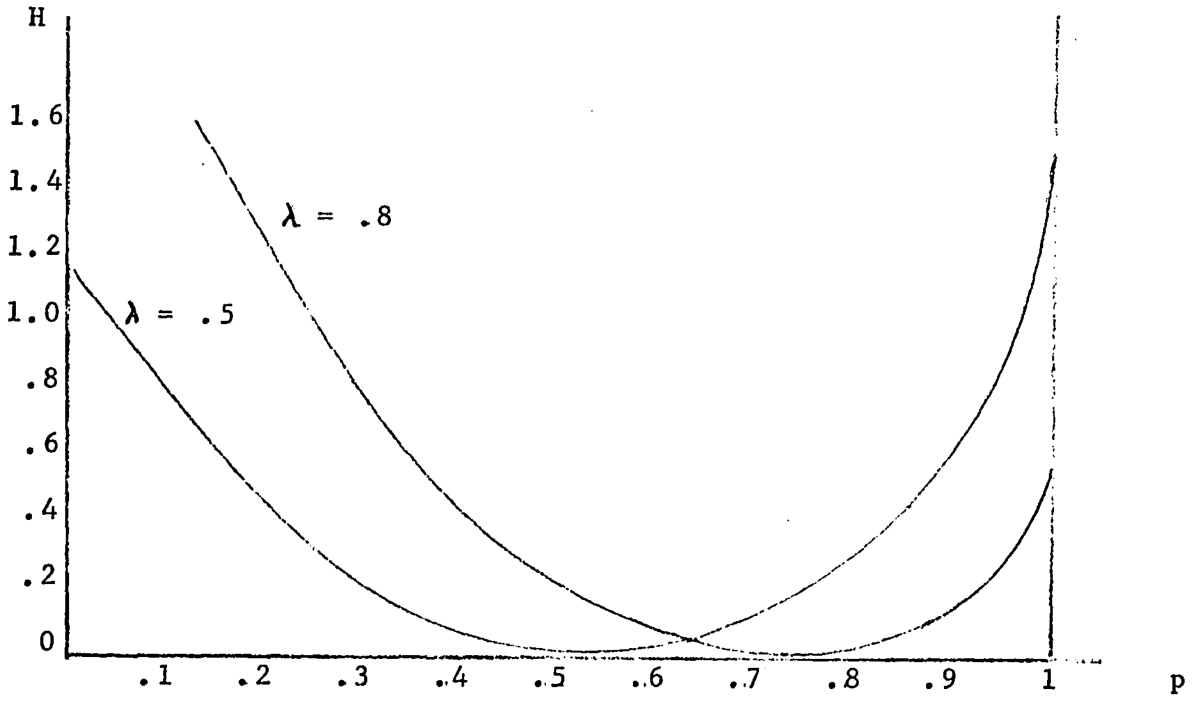


Figure 1.11 Excess bits for the binary case.

tropy in accordance with the channel status, hence avoiding excessive degradation of the data when the channel is overloaded.

The implementation of an adaptive data compression for multiple sensor outputs is given in (5); this reference also considers the effect of adaptivity on the ideal compression ratio.

For the simple case described above, it is clear that the upper bound of the bit compression ratio is (21)

$$(1.10) \qquad C_B \leqslant \frac{\log_2 M}{p \log_2(\frac{1}{p}) + (1-p)\log_2(\frac{M}{1-p})}$$

where p is the probability of making an accurate prediction.

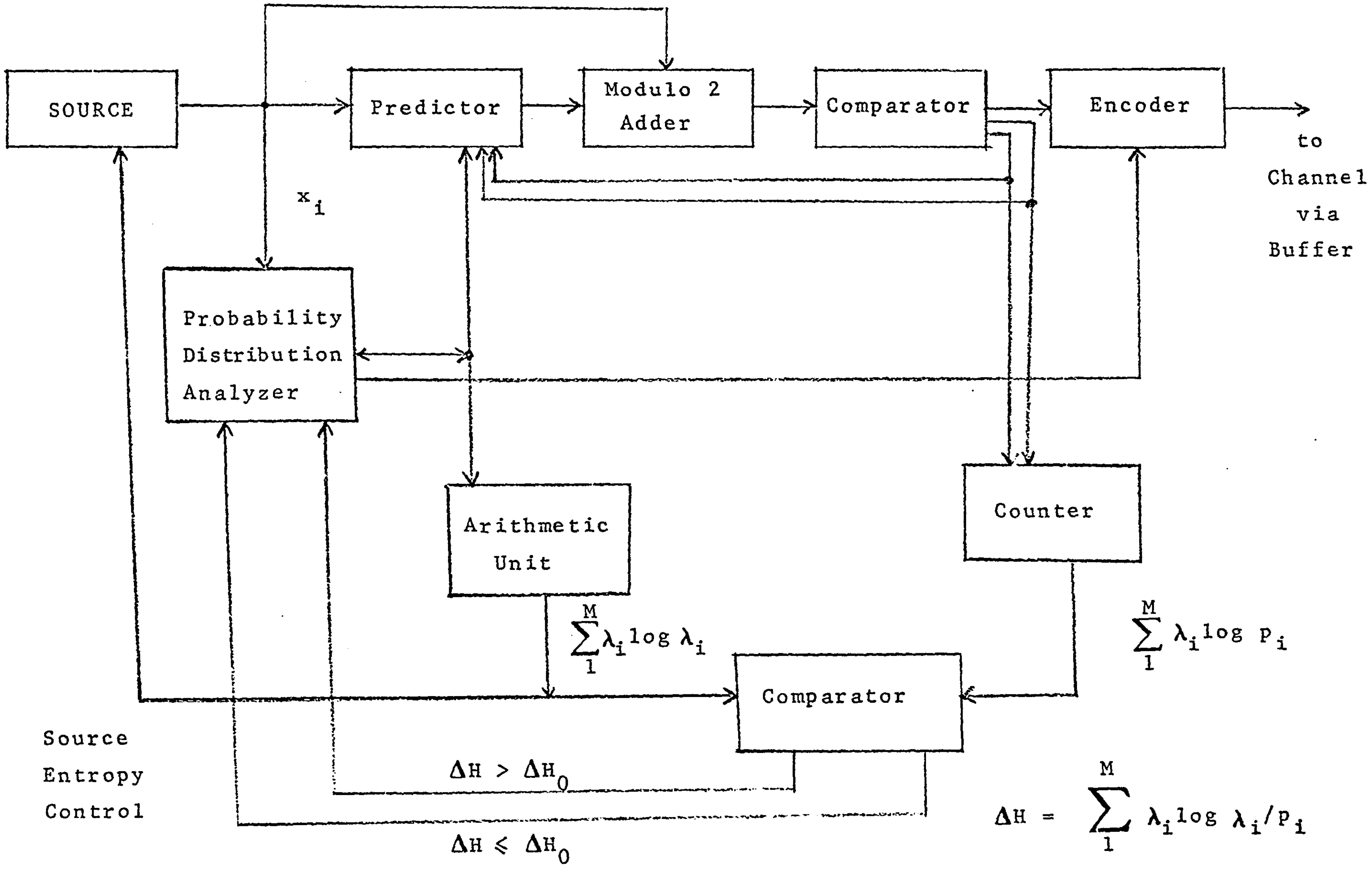

SOURCE
Predictor
Modulo 2 Adder
Comparator
Encoder
to Channel via Buffer
x_i
Probability Distribution Analyzer
Arithmetic Unit
Counter
$\sum_{1}^{M} \lambda_i \log \lambda_i$
$\sum_{1}^{M} \lambda_i \log p_i$
Comparator
Source Entropy Control
$\Delta H > \Delta H_0$
$\Delta H \leqslant \Delta H_0$
$\Delta H = \sum_{1}^{M} \lambda_i \log \lambda_i / p_i$

Figure 1.12 BLOCK DIAGRAM OF ADAPTIVE CODER

3. *Bit-Plane Encoding* This method consists in forming groups of M consecutive samples, and storing their quantized values vertically in a buffer memory. Each group contains, therefore, NM bits where N is the number of bits necessary to describe each sample. Looking at the memory as a rectangular array, we see that the least significant bits of all the words in the group lie on the same horizontal line. That is, the N^{th}-order bit of each sample is taken to form an M-bit sequence, called a "bit-plane. The bit-plane procedure is to encode and transmit the bits in each of the planes sequentially. It is obvious that when the data contain a high degree of predictability, the most significant bit-planes should contain long runs of zeros or ones. Thus these planes can be significantly compressed by some type of run-length encoding.

The following table describes the arrangement of bits in the memory, for M = 8 , N = 4 .

N \ M	1	2	3	4	5	6	7	8
4	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0
2	0	1	0	1	0	[1]	0	1
1	1	0	1	1	1	0	1	1

[1] is the second-order bit of the sixth word. The second-

order bit-plane is 0 1 0 1 0 1 0 1 .

The implementation of a bit-plane encoder is illustrated in Figure 1.13. The monitor looks at each bit-plane to determine hov each plane is to be treated. Essentially, the monitor distinguishes between four types of plane, depending on the "count-of-changes" C indicating the number of times adjacent bits in a bit plane are different. Thus, if $C = 0$, then the plane is monovalued and is described summarily (i.e., by transmitting only the value assumed by all the bits and indicating that the plane is monovalued). We refer to these planes as class A planes. Class B planes are those for which the number of runs is small enough to be profitably compressed, or more precisely, those planes for which $0 < C < (M/\log_2 M) - 1$. Run-length encoding could then be used. Finally, class C planes correspond to $C \geq M/\log_2 M$ and the plane is transmitted bit by bit.

The preceding plane identification can be accomplished with three bits. An extra bit could be used for single error detection. The best choice for the size M of a bit-plane depends on many factors and should be determined for each particular application.

We shall study the effect of channel noise on this technique in Chapter IV.

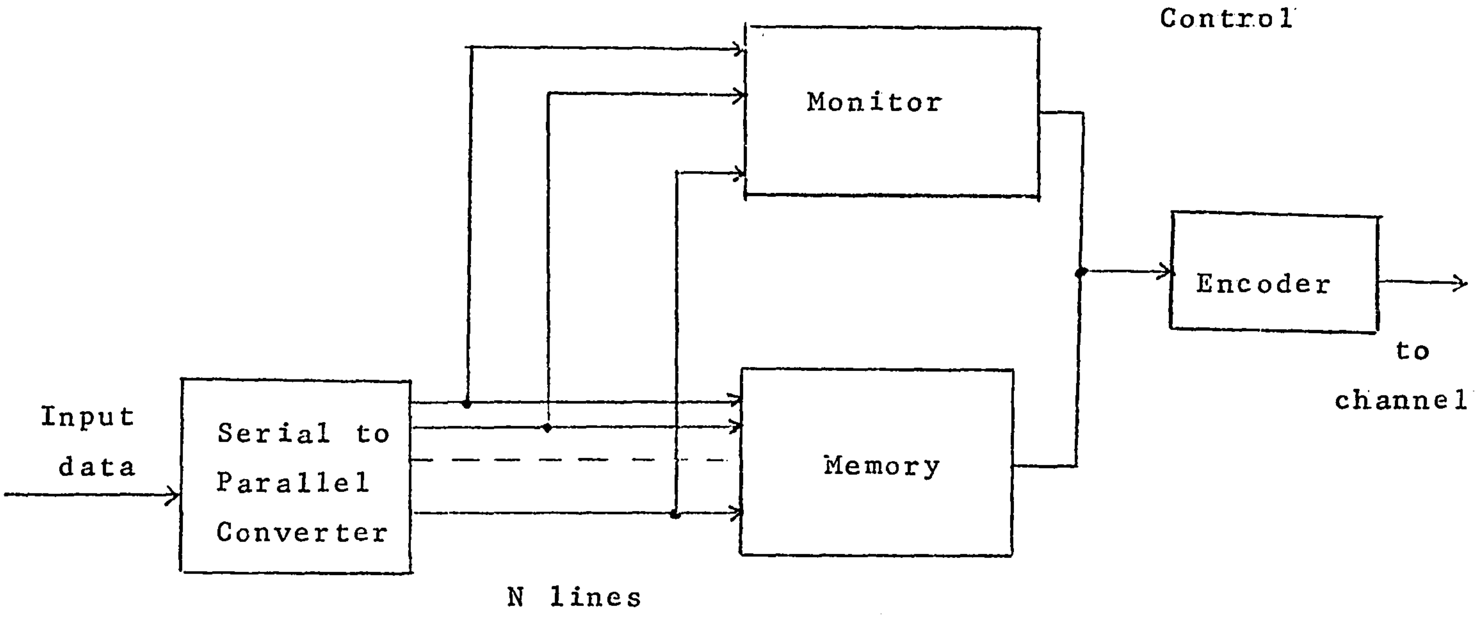


Figure 1.13 BIT-PLANE ENCODER

CHAPTER II

THE RANDOM VIDEO PROCESS

Compared to many forms of data transmission, picture transmission requires a relatively large bandwidth, since considerably more data seem necessary to produce an adequate visual signal.

Yet it has been known for a number of years that video signals exhibit a greater redundancy than any other information sources. This redundancy appears to the viewer in the following way; a large portion of the information conveyed by the picture seems to be concentrated in the contours of the objects rather than in their intensity. Since contours are determined by sudden variations of intensity, it is reasonable to assume that the video signal could be efficiently represented by difference signals, i.e.,"jumps" between adjacent samples. Large differences should occur less frequently than small ones since it is more likely that a randomly chosen point of an image lies in a "run" (i.e., an area of uniform or slowly varying intensity) than on a contour. This property suggests the use of differential PCM coupled with a Shannon-Fano coding procedure and some interesting results concerning DPCM appear in (20) and (21).

Another solution would be to take advantage of the existence of the uniform runs lying between contours; one could think of implementing a system which minimizes the number of bits describing those runs. The simplest method consists in a zero-order predictor associated with a run-length encoder, as described previously. Whatever the method used, the statistical correlation between neighboring elements sets a bound on the final efficiency (i.e.,compression) of the system. Compression techniques for a video source have been purely statistical, or psychophysical, or a combination of the two. In this study, we shall be mostly concerned with the former method, but some psychophysical properties will be briefly discussed.

This chapter reviews some of the most fundamental results obtained in picture bandwidth compression. It is divided in four sections; in the first section the statistical properties of a video source are described. The choice of appropriate coding techniques depends strongly on these properties. Their efficiency is usually measured against a standard which consists of PCM (uncoded) transmission. The second section, therefore, is concerned with PCM television and the effects of noise on the reconstructed image. Section 3 summarizes some forms of statistical coding and the last section describes some aspects of psychophysical coding.

2.1 Statistics of video signals

The large quantity of experimental data which have now been gathered to study the statistics of television signals allows us to judge the efficiency of the various models proposed up to date. The following results are mostly due to the work of Seyler

(22) , Franks (23), Estournet (24) and Kretzmer (25).

Any particular image can be modelled by a continuous function of three variables I(x,y,t) where x and y are the spatial coordinates and t is the time coordinate. The function I(x,y,t) represents the light intensity distribution of the image and can be written as

$$I = I (n\Delta x, m\Delta y, kT)$$

which corresponds to a discrete representation of the process. Note that sampling is always present in a TV signal and is due to line scanning which performs the mapping of a two-dimensional process into a function of time.

Investigations have been mostly concerned with first-order probability distribution of the levels of picture elements, the second and third order probabilities between adjacent picture elements and with the autocorrelation function of television signals. The main results are summarized below.

(i) Amplitude distribution is essentally non-stationary. Indeed different pictures yield different amplitude histograms and it has also been observed that even a single image can produce different histograms when certain photographic paramaters are varied (24). Since all forms of histograms can occur, one can state that the first-order probability of picture levels tends towards a uniform distribution if a sufficient number of frames are considered. Thus knowledge of the amplitude distribution alone gives no indication on the redundancy of the signal.

(ii) This redundancy becomes apparent when conditional probabilities are examined, i.e., when we consider the statistics of "difference" level between adjacent samples (elements); for a typical image the probability distribution of difference level is stationary and it is found that small differences are more probable than large ones. Moreover, the conditional probability of two adjacent samples having the same amplitude is about 10^6 times that of having amplitudes differing by the maximum amount. The distribution of sample differences is approximately laplacian.

(iii) From the statistical distribution of power in the frequency domain of the signal, we can deduce the correlations existing between elements in space and time (i.e., between the same spatial elements in successive frames). Franks (23) has shown that the autocorrelation function of the three-dimensional television process can be assumed separable, i.e., we can write

$$\phi(\tau) = h(\tau)\ g(\tau)\ f(\tau) \tag{2.1}$$

The three component functions $h(\tau)$, $g(\tau)$ and $f(\tau)$ represent the influence of element-to-element, line-to-line and frame-to-frame correlations, respectively. For typical picture material, there seem to be an extreme concentration of power near multiples of the line scan and frame scan rates.

A model characterizing the luminance process has been proposed by Franks who identifies the video signal with a random step function (Figure 2.1) with Poisson distributed zero crossings and independant amplitudes having a rectangular probability distribution.

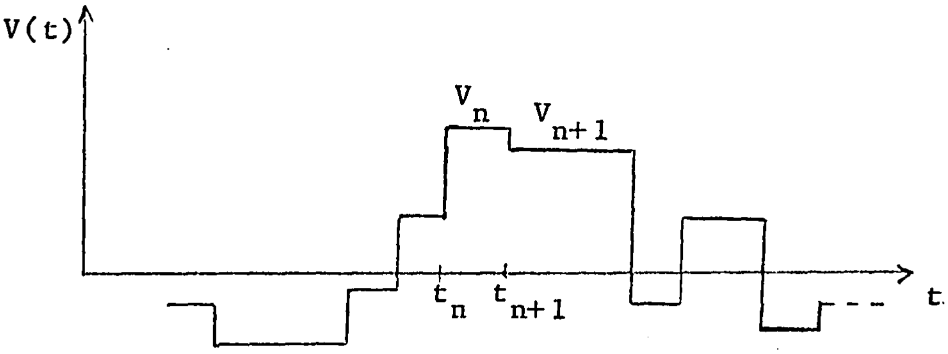


Figure 2.1 Random Video Signal

Assuming also that the random step function is a wide-sense stationary Markov sequence, it can then be shown that the correlation functions h(τ) and g(τ) are exponential.

Thus (2.1) becomes

$$(2.2) \qquad \phi(\Delta x, \Delta y, T) = A \exp(-\alpha|\Delta x| - \beta|\Delta y| - \lambda T)$$

where A is given by $A = \phi(0,0,0)$ and α, β, λ are constants. Equation (2.2) can also be written in the following form (considering spatial correlations only)

$$(2.3) \qquad \phi(\Delta x, \Delta y) = A \exp\left[-\sqrt{(\Delta x)^2 + (\Delta y)^2}\right]$$

Equation (2.2) has been reasonably confirmed by several correlation measurements and the average values of $\alpha = .0256$ and $\beta = .0289$ were derived empirically (26).(On the average $\alpha = \beta$)

The power spectrum of the process is given by the Fourier transform of (2.2) and power spectral density measurements performed by Deriugin (27) have also shown the validity of (2.2).

The existence of a non-zero autocorrelation function assures us that some compression is possible since, as shown by Elias (28), the autocorrelation function furnishes a lower bound to the redundancy of the signal. Thus for a high correlation A between neighbouring picture elements, the lower-bound redundancy is approximately equal to

$$R \cong -\frac{1}{2}\log_2(1 - A) \quad \text{bits/sample}$$

For typical picture material, the correlation between two points along the spatial dimensions was found to be of the order of .90 but this value decreases very rapidly with increase in the distance between samples.(Nyquist rate assumed).

Also measurements made on adjacent frames of motion picture films have resulted in a correlation factor of .80 .

Using the random step function described above to model the analog video process, Narayanan and Franks (29) have recently derived the power spectral density of digitally encoded video signals. The expression obtained consists of a product of four factors characterizing the effects of the digital pulse shape, quantizing and coding, scanning raster, and the bandwidth of the analog signal. It is also shown that large concentrations of power occur at multiples of frame rate, line rate and sampling rate when the video signal is encoded by PCM or DPCM. Various measurements have confirmed this periodic concentration of power.

In summary, the model proposed by Franks fits the statistical measurements of first and second order distributions. In particular this model permits us to have a better insight into the statistics of runs, which are analyzed next.

The transition matrix of the difference signal obtained from an image shows explicitely the dependence between levels and difference levels; the predictability of the signal lies in this dependence which manifests itself in the following way

(i) Successive jumps are likely to be of equal magnitude (or to differ by a small amount) if the first jump is not too large.

(ii) Given that the first jump is large, it will more likely be followed by a small jump.

Note that this process is upper-bounded since the signal has finite amplitude. However the probability of no jump (zero difference) between two picture elements is larger and implies the existence of large picture areas (runs) where levels are constant within a small tolerance η.. We are interested here in the statistical behaviour of the run lengths. We know that the amplitudes S_j of the samples are dependent. Consider a run consisting of n samples $\left\{S_{ij}\right\}$, (j = 1, ... ,n). One can assume that the amplitudes of runs $\left\{S_{ij}\right\}$, (i = 1,2, ...) are stationary and that they occur randomly and independently in time (the level difference between runs consisting of a dependent random variable with an exponential probability distribution).

We are interested in obtaining the epected value of the run length, i.e., we want to compute

$$E\ (w) = \int_0^\infty w\ p(w)\ dw$$

where w_i ($i = 1,2, \ldots$) is the random variable denoting the run length and p(w) is the probability density of the run length.

Now we have assumed that the distance from an arbitrary point t_0 to the next random point t_i is a r.v. independent of what happened outside the interval (t_0, t_i). But this is equivalent to

$$p\ (\ w/\ w \geq t_0\ ) = p\ (\ w - t_0)$$

It is shown in probability calculus that the only function satisfying the above condition is an exponential. Furthermore, since the autocorrelation function of the video process is given by

$$\phi(\tau) = \lambda e^{-\lambda\tau}$$

it can be shown that

$$p(w) = \lambda e^{-\lambda w}$$

and the probability distribution of the run length is therefore given by

$$P(w) = \int_0^w \lambda e^{-\lambda z}\, dz = 1 - e^{-\lambda w}$$

Since we are supposing that the video signal is sampled at a fixed rate, we must define a Poisson process for a quantized system, i.e., the r.v. w_i can assume discrete values only; in other words we consider the case where the occurence of runs takes place at times n Δt, where n is an integer and Δt is the minimum duration of the run. For convenience we set $\Delta t = 1$. The minimum run length is therefore composed of one sample at least, and to obtain the discrete probability density p(n), the area under the continuous density function p(w) between points w = (n-1) and w = n , is lumped at point n. Thus

$$\text{(2.4)} \qquad p(n) = P(w \leq n) - P(w \leq n-1)$$

$$= (1 - e^{-\lambda n}) - (1 - e^{-\lambda(n-1)})$$

$$= (e^{\lambda} - 1)\, e^{-\lambda n}$$

where $1 \leqslant n < \infty$.

This expression represents the probability distribution of run length in discrete form, and the expected value of run length is given by

$$E(n) = \sum_n n\,(e^{\lambda} - 1)\, e^{-\lambda n}$$

$$\text{(2.5)} \qquad = (e^{\lambda} - 1) \sum_n n\, e^{-\lambda n}$$

Noting that

$$n\, e^{-\lambda n} = - \frac{d}{dn}(e^{-\lambda n})$$

and that

$$\sum_n e^{-\lambda n} = \frac{1}{1 - e^{-\lambda}}$$

we obtain from (2.5)

$$E(n) = (e^{\lambda} - 1) \frac{e^{-\lambda}}{(1 - e^{-\lambda})^2} = \frac{1}{1 - e^{-\lambda}}$$

The probability distribution of run length given by (2.4) can therefore be written as

$$\text{(2.6)} \qquad p(n) = \frac{1}{E(n) - 1} \left(\frac{E(n) - 1}{E(n)} \right)^n$$

This result is in agreement with the statistical measurement of run length performed by Cherry (31).

It is sometimes convenient to write Equation (2.6) in the following way

$$p(n) = \frac{p(1)}{1 - p(1)} \left(1 - p(1)\right)^{n} \qquad (2.7)$$

where p(1) is the probability of a run consisting of a single element.

Considering now a jump ξ of amplitude $\frac{2^{x} - 1}{2^{k}}$ (where x varies from 0 to k), it is possible to derive the relation which exists between the statistics of runs and those of jumps. Indeed, it can be shown that (21)

$$1 - p(1) = \frac{1}{2^{x}} \sum_{i=0}^{2^{x}-1} p^{2x}(i/i)$$

where p(i/i) is the probability of a sample with amplitude i given that the preceding sample has the same amplitude.

2.2 PCM Encoding of Video Signals

Before compressing digitally encoded video signals, it is important to know the effects of changes in system parameters on the picture quality when PCM transmission is used. Given a certain picture quality, the designer wishes to choose the system parameters such that the number of bits per frame to be transmitted is a minimum. The effect of sampling and quantizing on picture quality has been studied by many authors, and this section is based mainly on the works of T.S. Huang (30), R.E.Graham (32), Seyler(33) and Roberts (34).

2.2.1 Visibility of Noise

Visual response to noise is an important factor of any picture coding system, since noise is due both to channel imperfections (additive Gaussian noise), and the encoding procedure, which yields an output within some finite error. For example, the finite number of quantization steps of PCM produces what is usually called the quantizing noise. However, it is important to predict noise visibility under a wide variety of conditions.

What is known can be summarized in the following facts (35)

1) Noise is less visible in a complicated picture.

2) Noise is more visible if it is correlated with the picture than if it is random. Hence, quantizing noise is more visible than additive random noise of the same r.m.s. value.

3) The presence of noise in a picture reduces its contrast and its sharpness.

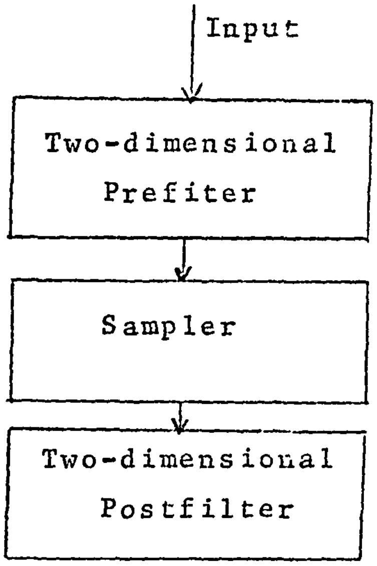
4) The spectrum of the image affects the visibility of noise in a way which is not yet fully understood.

5) Randomly scattered noise is usually less visible than noise with local structures (i.e., noise which occurs in bursts).

The quality of the received picture is clearly affected by the visibility of noise and other distortion occurring during the transmission. The influence of various system parameters on the noise visibility has been investigated, and we outline next some of the results.

2.2.2 Sampling

Consider first the case of a noiseless channel and assume L samples are taken to describe a line. Each sample is quantized into one of 2^k distinct levels. For commercial television, about 500 samples per line (l = 500) and 50 to 120 brightness levels (k = 6 or 7)* are requiree to achieve a resolution comparable to present-day analog system. A smaller L results in poorer resolution while smaller k introduces artificial contours. The total number of bits per picture is N = L L k. Consider now the following sampling process (32)



Peterson and Middleton (36) have shown that for a fixed number of samples per frame, prefiltering and postfiltering with ideal low-pass filters yield the least m.s. difference between the output and the input. Subjective tests performed by Huang have

* This is true if uniform quantization is used, However, one bit can be saved by using logarithmic quantization, thus matching more closely the properties of human vision.

consolidated that theory.

Huang (37) has also shown that the sampling pattern affects the output picture quality. Moreover, he showed that, given N, the total number of bits per picture, there seems to be an optimum choice for the values of L and k. From a series of subjective tests, isopreference curves were drawn (the points on these curves represent pictures of equal subjective quality, for various values of L and k), indicating strong dependence on the picture type. Hence, for pictures with a large amount of details, k can be small (only a few brightness levels are needed), but L should be large. In this case, the optimum value is picture dependent. In general L should be large for a picture with a large amount of detail, while in a picture with a small amount of detail, k should be large.

2.2.3 Quantization

Quantization noise can be reduced by placing a prefilter and a postfilter around the quantizer, and D.N.Graham (38) has obtained with this method a picture essentially free of artificial contours using only three bits per sample. Generally, a smaller number of quantization levels can be used if the quantization noise can be transformed into random noise. An interesting technique has been proposed by L.G.Roberts (34). It consists in a pseudo-random noise modulation technique, in which a noise with rectangular spectrum and peak-to-peak value equal to one quantum step is added to a picture before quantization, the same noise being substracted from the quantized received picture. Roberts showed that this procedure would result in an unquantized output to which has been added a random noise with the same r.m.s. value. With this method, four bits per sample have been found acceptable.

2.2.4 Coding and Channel Noise

For a noiseless channel, the particular code chosen to represent the 2^k brightness levels has no bearing on the received picture. However, in the case of a noisy channel, the amount of noise in the received picture depends on the code chosen. For the k-bit straight binary code, the noise power is given by*

$$N_B = \frac{P_B}{3}(2^{2k} - 1)$$

where P_B is the channel error probability, and for a k-bit reflected binary Gray code, it is equal to

$$N_b = (4^k - 1)/6 - \frac{1-2P_B}{2}\,\frac{4^k - (1-2p)^n}{4 - (1-2p)}$$

It is clear that

$$N_b \geqslant N_B \qquad \text{for} \quad P_B \leqslant 1/2$$

Thus a Gray code results in a larger average noise power.

Generally speaking, the subjective effect of noise is not the same for digital transmission and analog transmission. It has been found (37) that for high SNR, white Gaussian noise is more annoying than the noise arising from a binary symmetrical channel, while for low SNR the reverse is true, the crossover point being about 20 dB.

* This expression is derived in Chapter III (Section 3.1)

2.3 Psychophysical Coding

The coding methods proposed in Section 1.1 all rely on the statistical constraints which exist among the picture elements. To improve the efficiency (in terms of compression ratio) of picture coding, one can take advantage of the properties ot human vision. In other words, the fidelity criteria are matched more closely to the psycho-physics of vision. Thus psychophysical coding consists in altering the original picture in such a way that it can be described by a smaller number of bits. Note that this process is irreversible since what has been discarded from the original signal cannot be recovered. This distortion of the signal should, however, not be noticeable by the human viewer. Briefly, what is discarded is what would not have been seen anyhow. Experiments conducted at the M.I.T. and by Seyler (39) have confirmed the following facts.

- In vision, spatial and contrast resolutions are exchanged so that the number of contrast levels that can be distinguished in small objects is substantially smaller than in large, almost uniform areas. Thus, if run length encoding is used, the level of short runs could be quantized more coarsely than that of long runs, resulting in a smaller overall bit rate.

- Motion resolution is also exchanged for detail resolution, i.e., the resolution of spatial details in moving objects deteriorates. Hence, the number of samples per frame could be reduced.

- Spatial resolution is considerably reduced when the observer is confronted by a sudden change of scene. Extensive tests performed by Seyler (39) have shown that the human observer would not perceive a temporary reduction of spatial detail for an average of 750 milliseconds after a scene change. The experiment consisted in reducing temporarily the bandwidth of standard television signals after scene change by means of a transient controlled low-pass filter. It was found that the initial bandwidth could be set at one twentieth of the system bandwidth reached at the end of the recovery transient.

This result is particularly useful when frame difference coding is applied. This method takes advantage of the statistical correlation which exists between adjacent frames (40) and results in what Seyler has called frame run coding. The coding procedure consists in transmitting a new frame only when consecutive frames display a sufficient number of different elements. However, to obtain a reasonable compression ratio with this method, it would be necessary to take large averaging intervals in view of accomodating scene change. This drawback can be overcome by subjecting the picture to a resolution transient when the scene changes. During the first quarter of a second, only 20% of the total samples need be sent, thus reducing the bit rate even under sudden and complete scene change in the flow of pictures.

All forms of coding described above have a common property, which becomes more apparent in a noisy channel. It consists in the fact that an error occurring in one of the coding words propagates beyond the time of occurrence of this error. In other words, there is a shift in the position of all data following the error. This shift usually persists until a sync pulse resets the system. In the case of television signals, the length of the data between sync pulses constitutes a line; hence error will propagate over an entire line. In the next chapter, we compute the magnitude of this error.

CHAPTER III

EFFECT OF CHANNEL NOISE ON ZERO-ORDER PREDICTOR

We have seen in Chapter I that the standard by which to compare data compression techniques is taken as a constant rate, time-sampling PCM system. Therefore, we first determine the effect of noise on the reconstructed waveform in a PCM system. We assume that the channel is corrupted by additive white Gaussian noise with a zero mean and one-sided spectral density N_0.

3.1 PCM Systems

Let the amplitude of the source be uniformly distributed between 0 and +1 volt, and suppose that each sample is quantized into q levels as shown in Figure 3.1.

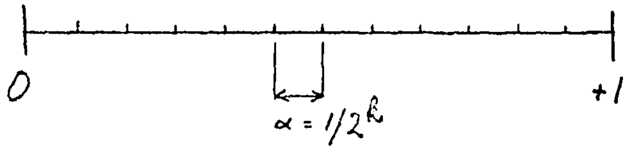


Figure 3.1

Then $q=2^k$, where k is the length of the PCM word. The mean square (m.s.) error ϵ^2_{PCM} in the reconstructed data can be expressed as the sum of three independent errors. Thus,

$$\epsilon^2_{PCM} = \epsilon^2_1 + \epsilon^2_2 + \epsilon^2_3$$

where

ϵ_1^2 expected squared quantization error,

ϵ_2^2 expected squared transmission error,

ϵ_3^2 expected squared threshold error.

The error due to quantization is easily found to be

$$(3.1) \qquad \epsilon_1^2 = \int_{-\alpha/2}^{\alpha/2} \alpha\, x^2 dx = \frac{1}{12\ 2^{2k}}$$

where $\alpha = 1/2^k$. This error is independent of the channel characteristics and the modulation scheme employed, and represents a lower bound for the overall error ϵ_{PCM}^2. We regard it as an implementation error since it depends only on the number of bits, k, per word which is initially chosen by the designer. Thus the quantization error is common to both compressed and non-compressed systems.

ϵ_2^2 is due to channel noise; we assume that P_B, the probability of bit error is small enough that only one error need be considered in each word consisting of a sequence of k bits, and that the probability of a given bit in error is uniform over the length k of the word. We may then write

$$\epsilon_2^2 \,/\, \begin{matrix}\text{error in a}\\ \text{PCM word}\end{matrix} = \frac{1}{k} \sum_{j=1}^{k} 2^{-2j}$$

The probability of one error in a sequence of k bits is

$$\Pr(\text{one error}) = k\ P_B\ (1 - P_B)^{k-1} \cong k\ P_B$$

Thus the transmission error ϵ_2^2 is given by

$$(3.2) \qquad \epsilon_2^2 = \frac{k\ P_B}{k} \sum_{j=1}^{k} 2^{-2j} = \frac{P_B}{3}(1 - 1/2^{2k})$$

Note that this relation holds for straight binary encoding only.

Karp (41) has shown that (3.2) holds even when we consider the possibility of more than one error in a k-bit word.

3.2 Bit Error Probabilities

The expression (3.2) found for the m.s. error of a PCM system depends on the parameter P_B, the bit error probability. Theoretical expressions have been derived for bit error probabilities; essentially P_B depends on the mode of transmission (i.e., the modulation technique used at the transmitter), the propagation medium (fading or nonfading) type and the detection technique used at the receiver. Table III-1 summarizes some values of P_B.

In this study we shall concentrate on the following scheme; matched filter-coherent detection for nonfading medium. We also assume binary antipodal signal.

We define the signal power to be S watts; E is the signal energy per bit, and the data rate is R bits/sec. Then for binary transmission $T = 1/R$, and we have

$$E = S\ T = S/R$$

SIGNALLING METHOD	Receiver Characteristics	Nonfading Medium	Fading Medium
FSK (Frequency shift keying)	Matched Filter-Coherent Detection	$\frac{1}{2}(1 - \text{erf } E/N_0)$	$\frac{1}{2}\left[1 - \left(\frac{E/N_0}{E/N_0 + 2}\right)^{\frac{1}{2}}\right]$
FSK	Matched Filter-Incoherent Detection	$\frac{1}{2} e^{-E/2N_0}$	$\frac{1}{E/N_0 + 2}$
PSK (Phase shift keying)	Matched Filter-Coherent Detection	$\frac{1}{2}(1 - \text{erf } E/N_0)$	$\frac{1}{2}\left[1 - \left(\frac{E/N_0}{E/N_0 + 1}\right)^{\frac{1}{2}}\right]$
DPSK (Differential Phase shift keying)	Matched Filter-Differentially Coherent Detection	$\frac{1}{2} e^{-R/N_0}$	$\frac{1}{2}\left(\frac{1}{E/N_0 + 1}\right)$

Table III-1 BIT ERROR PROBABILITIES

Note The information in this table is to be found in

M.Schwartz, W.R.Bennett, S.Stein "Communication Systems and Techniques"

Then with coherent reception, P_B is minimized for antipodal signals and is given by

$$P_B = \text{erfc}\sqrt{\frac{2E}{N_0}} = \text{erfc}\sqrt{\frac{2S}{N_0 R}} \; .$$

We can now ,following Viterbi (42), express P_B in terms of S/N_0B , the channel SNR in the bandwidth of the modulation, and q, the number of quantization levels.

If the sampling period is τ , we have

$$B = \frac{1}{2\tau}$$

But, if each sample is quantized into 2^k levels, k bits must be sent every τ seconds. Hence

$$\tau = kT = k/R = \frac{\log_2 q}{R}$$

and

$$B = \frac{R}{2 \log_2 L} \; .$$

Thus we obtain for coherent reception

$$P_B = \text{erfc}\sqrt{\frac{2S}{N_0 R}} = \text{erfc}\sqrt{\frac{S/N_0 B}{\log_2 q}} \; .$$

We now derive a useful approximation due to the asymptotic expression for the complementary error function.

$$\text{erfc } x = \frac{2}{\sqrt{\pi}} \int_0^\infty e^{-y^2} dy \cong \frac{e^{-x^2}}{x\sqrt{\pi}} \sim e^{-x^2} , \quad x \gg 1.$$

where $\sim$ indicates an order of magnitude.

Hence, for coherent bipolar transmission we have

$$(3.3) \qquad P_B = \mathrm{erfc}\sqrt{\frac{2S}{N_0R}} \sim \exp\left(- \frac{2S}{N_0R} \right)$$

where only the dominant exponential factor has been retained. In particular, this expression shows clearly the effect of the transmission rate on P_B. Suppose we know P_B for a given R. Then if we wish to transmit information at a lower rate, say $R' = R/\alpha$, we obtain the error probability

$$(3.4) \qquad P_B' = \exp\left(- \frac{2S}{N_0R/\alpha}\right) = \left[\exp\left(- \frac{2S}{N_0R}\right)\right]^{\alpha} \sim P_B^{\alpha}$$

If $P_B \ll 1$, the bit error probability for the slower rate can be many order of magnitude smaller than P_B.

This is a fundamental result in digital communication that has been often neglected when data compression is considered.

3.3 Data Compression Systems

Similar to the PCM systems, data compression systems are subjected to both quantization and transmission errors, but the output of a compressor can also be further degraded by an implementation error caused by the tolerance chosen to compare the data sample at the output of the fixed-rate sampler. We call this type of error "aperture error" and compute the resulting m.s. error.

3.3.1 Aperture Error

For a zero order system, each sample is declared either redundant or nonredundant, depending on the relative value of the actual sample with respect to the value of the preceding sample. Denote the aperture magnitude by γ and let S_i be the amplitude of the i^{th} sample.

Then, if

$$S_i - \gamma \leqslant S_{i+1} \leqslant S_i + \gamma$$

S_{i+1} is redundant and is not sent.

If $S_{i+1} - S_i > \gamma$, i.e., if the sample S_{i+1} value falls outside the $S_i \pm \gamma$ bounds, then it is nonredundant and must be transmitted.

Let a denote the magnitude of a quantization step, and suppose that the signal has been quantized before entering the data compressor. Then the amplitude difference between two signals are multiples of a and we can express the aperture γ as a function of a

$$\gamma = \pm ma \quad , \quad m = 0, 1, 2, \ldots$$

If we set $m = 0$, (i.e., a sample is nonredundant unless it exactly equals the preceding sample) there is no aperture error and the m.s. error of the reconstructed waveform depends only on the quantization error (defined in preceding section) and the transmission errors.

The error introduced by an aperture of ma is then $\pm a, \pm 2a, \pm 3a, \ldots, \pm ma$ and we assume that it is uniformly distributed in the interval $(-ma, \ldots, 0, \ldots, +ma)$.

Hence

$$P(\gamma_i) = \frac{1}{2m+1}$$

where $\gamma_i = ia$, $i = 0, \pm 1, \ldots, \pm m$.

The mean square aperture error for a redundant sample is then

$$(3.5) \qquad E(\gamma^2) = \sum_{i=-m}^{m} P(\gamma_i)\, \gamma_i^2 = \frac{1}{2m+1} \sum_{i=-m}^{m} a^2 i^2 .$$

$$= \frac{2a^2}{2m+1} \sum_{i=1}^{m} i^2 = \frac{a^2 m(m+1)}{3}$$

But $a = 1/2^k$, where k is the length of the quantized word. Also for a nonredundant sample $E(\gamma^2) = 0$ since the actual value of the sample is transmitted. Errors will occur only in $(C_{sm}-1)$ redundant samples and the m.s. aperture error for the reconstructed waveform is thus

$$(3.6) \qquad \gamma^2_{\text{aperture}} = \frac{m(m+1)}{3(2^{2k})} \left(\frac{C_{sm} - 1}{C_{sm}} \right)$$

where C_{sm} is the average sample compression ratio measured when the compressor has an aperture m (i.e., the compression ratio is a function of the aperture).

3.3.2 Transmission Errors

Transmission errors (due to white Gaussian noise) in data compression systems fall in two categories; namely, the errors in level informations and that in timing informations.

Errors in level information

We consider first an asynchronous compression system; then no timing information need be sent and errors occur in the level words only. However, in this case, an error affects an average of C_s words. This propagation of the error over C_s

samples can therefore be expressed in function of the transmission error of a non-compressed system, and referring to Equation (3.2) we have

$$(3.7) \qquad \epsilon^2_{\substack{\text{compressed}\\ \text{asynchronous}}} = C_s\,\epsilon_2^2 = C_s \frac{P_B}{3}\left(1 - 1/2^{2k}\right)$$

where $P_B = \exp(-\frac{2S}{N_0 R})$ is the error probability of the non-compressed PCM system. Note that here $C_s = C_B$ since no addressing scheme is used.

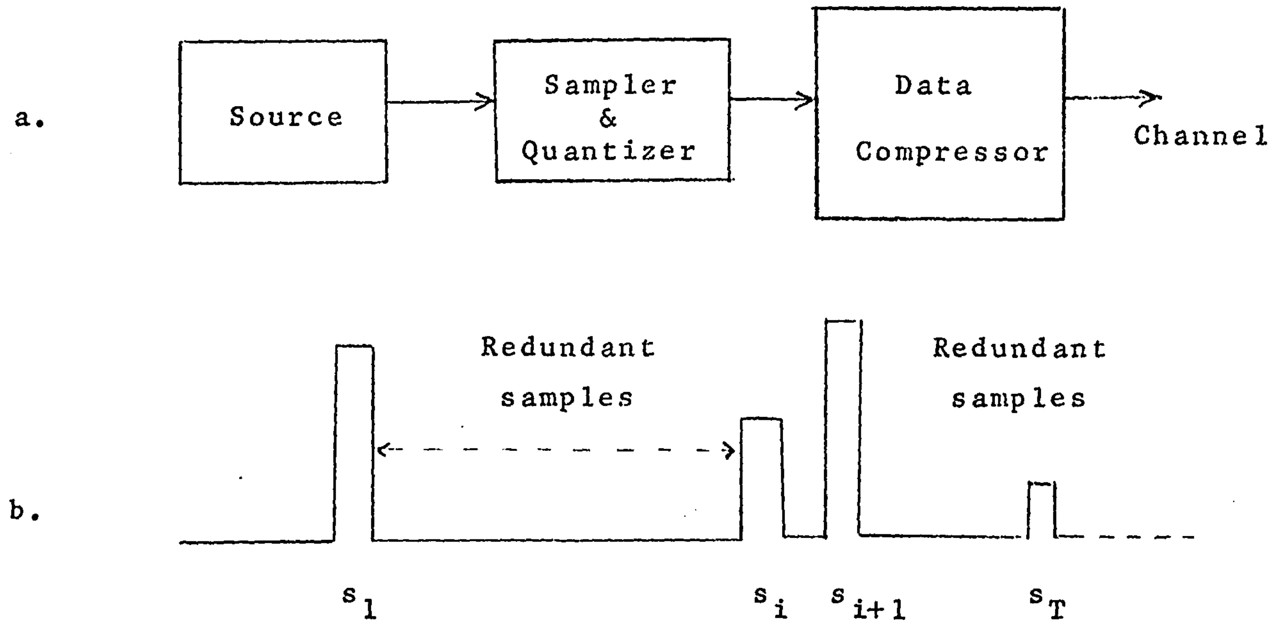


Figure 3.2 a. Asynchronous Data Compression

b. Possible sample pattern sent through the channel

Referring to Figure 3.2-a, it is evident that an asynchronous system does not result in bandwidth compression. In the pattern of samples shown (one of many possible), samples s_i and s_{i+1} are adjacent and nonredundant, hence they appear at the compressor output as they would in a PCM system. The bandwidth of transmission in an asynchronous system, being determined by the time interval between the "closest" significant samples, must therefore be the same as the PCM bandwidth if the pulses representing s_i and s_{i+1} are to be transmitted with no additional distortion.

Thus, although the average rate of the asynchronous system is smaller than the rate of the non-compressed system, the bit error probability P_B must depend on the highest rate which could possibly appear during the transmission of the whole message. Clearly, the maximum rate is that of the non-compressed system, i.e., R.

The m.s. error due to transmission errors in an asynchronous compression system is C_s times the m.s. error of a non-compressed system. To improve the performance of the compression system, we can

a) increase the quantization resolution, hence increase k,

b) decrease P_B by increasing the signal energy,

c) use coding tachnique by adding back some controlled redundancy.

Note that the three methods result in a smaller overall compression ratio, i.e., C_B departs from C_s.

The first two methods are analyzed next.

(a) Suppose we increase the quantization resolution by x bits. Then the word length becomes (k+x) bits and the m.s. error due to quantization is now

$$\epsilon_1^2 = \frac{1}{3\ 2^{2(k+x)}} .$$

For a word length of k bits, the m.s. error is

$$\epsilon_1^2 = \frac{1}{3\ 2^{2k}}$$

Hence, denoting by $\Delta\epsilon_1^2$ the decrease in ϵ_1^2 resulting from the use of (k+x) bits per word, we can write

$$\epsilon_1^2 = \frac{1}{3\ 2^{2k}} - \frac{1}{3\ 2^{2(k+x)}} = \frac{2^x - 1}{3\ 2^{2(k+x)}}$$

But, the increase in the bit rate causes the following

(i) A decrease in compression ratio. Indeed, with a word length of (k+x) bits, we have

$$C_B = \frac{C_s}{1 + x/k}$$

(ii) An increase in the bit rate R. The new rate is now

$$R' = \frac{k+x}{x} R .$$

(iii) An increase in the m.s. error due to transmission errors, i.e.,

$$\epsilon_2^2 = \frac{P_B'}{3} (1 - 1/2^{2(k+x)}) C_s . \tag{3.8}$$

Note that the bit error probability is increased since we now have

$$P_B' = P_B(\frac{k}{k+x})$$

The loss incurred by this method is thus

$$(3.9) \qquad \Delta\epsilon_2^2 = \frac{P_B'}{3}(1 - 1/2^{2(k+x)})\ C_s - \frac{P_B}{3}(1 - 1/2^{2k})$$

The problem is now, given P_B, k and C_s, what is the value of x which yields

$$(3.10) \qquad \Delta\epsilon_1^2 \geqslant \Delta\epsilon_2^2 .$$

This value of x must then be inserted in (1.2) to obtained the bit compression ratio and see if C_B is large enough to make compression still worthwhile.

By varying k through a wide range of values (2 to 11) and computing $\Delta\epsilon_2^2$ and $\Delta\epsilon_1^2$, it is seen that inequality (3.10) is satisfied for x = 1. In this case C_B is given by

$$C_B = \frac{k}{k+1}\ C_s$$

(b) We assume here that the value of P_B is such that the quantization error is the dominant term in the equation for the m.s. error $\epsilon^2_{\text{non-compressed}}$. We wish to determine the increase in signal energy necessary to obtain

$$(3.11) \qquad \epsilon^2_{2_{\text{compressed}}} = \epsilon^2_{2_{\text{PCM}}}$$

Referring to (3.8) we see that the value of P_B must decrease to $P_B' = P_B/C_s$; but

$$P_B' \sim \exp(- \frac{2S}{N_0 R})^{\alpha} \sim P_B^{\alpha}$$

Therefore we must have

$$P_B^{\alpha} \sim P_B/C_s$$

or

$$\exp(\alpha \log P_B) \sim P_B/C_s$$

and

$$\alpha \log P_B \sim \log P_B - \log C_s$$

yielding

$$\alpha = 1 - \frac{\log C_s}{\log P_B} = 1 + \frac{\log C_s}{\log 1/P_B} > 1$$

Hence

$$P_B' = \exp\left(-\frac{2S\left[1 + \log C_s/\log(1/P_B)\right]}{N_0 R}\right) \tag{3.12}$$

For small C_s and $P_B \ll 0.5$ the increase in signal power is small and the choice between method (a) or (b) will depend on the particular application of the system and the available transmitter power.

We now consider the effect of transmission errors in level words on synchronous compression systems.

Figure 3.3-b illustrates a typical set of samples as it would appear at the output of the compressor. In an asynchronous system, the samples would be fed directly into the channel. To obtain bandwidth compression, the samples are buffered and read out at a fixed rate into the channel. The effect of buffering, as can be seen in Figure 3.3-b is to eliminate all indication pertaining to the location of the nonredundant samples; hence timing words must accompany the level words, thus enabling the receiver to restore the non-transmitted samples. The bit compression ratio C_B depends on the scheme used to transmit this

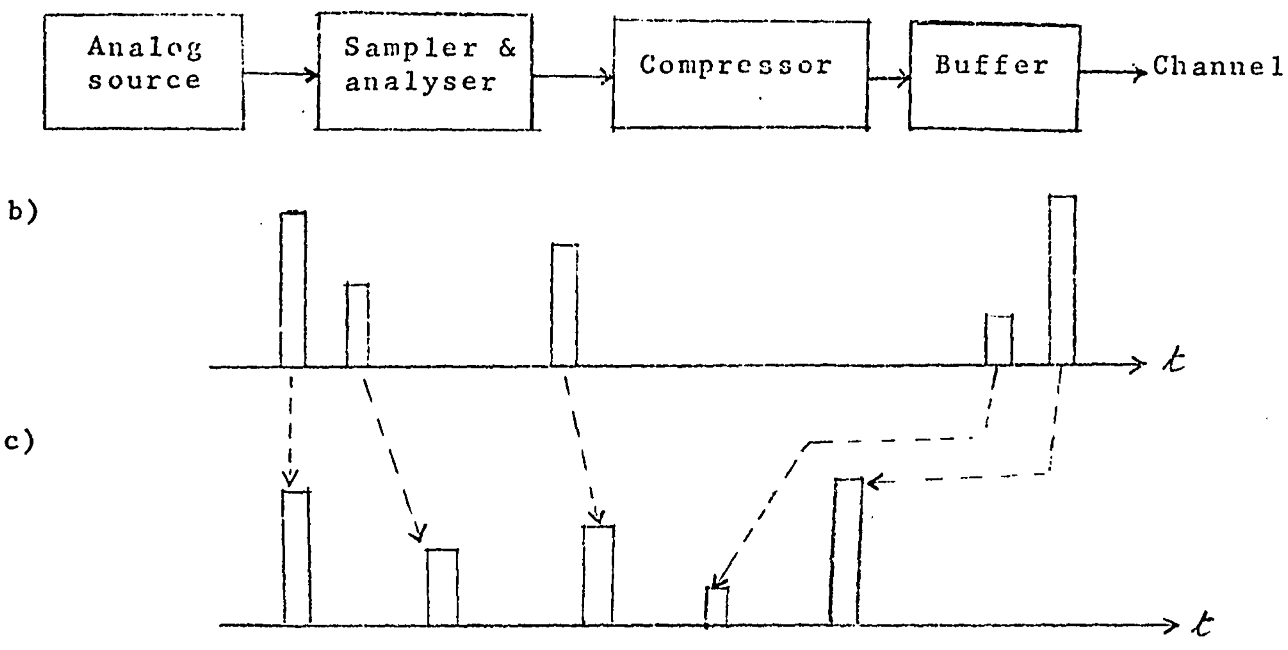


Figure 3.3 a. Synchronous Compression System

b. Pattern of samples at output of compressor

c. Same information at output of buffer

extra information and determines the rate of the compressed data. As in the asynchronous case, bit errors propagate over an average of C_s samples, and since the rate of transmission is now $1/C_B$ that of the non-compressed system, we may write

$$(3.13) \qquad \epsilon^2_{2_{\text{level}}} = \frac{C_s}{3}\,(1 - 1/2^{2k})\,(P_B)^{C_B}$$

The total m.s. error due to channel noise is

$$\epsilon^2_2 = \epsilon^2_{2_{\text{level}}} + \epsilon^2_{2_{\text{timing}}} .$$

We shall determine $\epsilon^2_{2_{\text{timing}}}$ in the following paragraph. We observe that $\epsilon^2_{2_{\text{level}}}$ increases linearly with C_s and decreases exponentially with C_B. Hence, for small P_B and close values of

C_s and C_B, $\epsilon^2_{2_{\text{level}}}$ is smaller than the ϵ^2_2 of the non-compressed system. We have

$$C_B \geqslant 1 - \frac{\log C_s}{\log P_B} \qquad \epsilon^2_{2_{\text{level}}} \leqslant \epsilon^2_{2_{\text{non-compressed}}}$$

$$C_B < 1 - \frac{\log C_s}{\log P_B} \qquad \epsilon^2_{2_{\text{level}}} > \epsilon^2_{2_{\text{non-compressed}}}$$

Errors in Timing Information

We will analyse here the effect of bit error in the address word for zero order systems, and consider first the case of run length encoding.

A. Run Length Encoding

This addressing technique identifies the position of the data elements along a line by keeping a running total of all received run lengths. Let L be the number of elements in a line (recall that a line is framed by synchronous pulses which are assumed error-free).

Then the position of the n^{th} transmitted sample will be in error if any run length word $(RL)_i$, $(1 \leqslant i \leqslant n)$, is in error. Also, if an error is made in $(RL)_n$, then the position of all samples S_m where $n \leqslant m \leqslant L$ will be in error, i.e., errors in position propagate to the end of the line. Hence, the affected reconstructed samples are shifted in position by an amount equal to the error. Consider, for example, an error in the least significant bit of $(RL)_n$. This will cause a displacement of ±1 sample for the remaining segments, that is, only one sample per n reconstructed samples is affected by this error. Also, assuming that the run length word consists of r bits, an error in the most significant bit of $(RL)_n$ will cause a displacement of $\pm 2^{r-1}$ samples for the remaining segments, thus resulting in a loss of synchronization for the remainder of the line.

Note that when C_s is the average compression ratio per line, then exactly C_s run-length words must be sent per line* and we can take the number of nonredundant samples per line to be L/C_s (a fixed number).

If, however, C_s has been obtained by averaging the redundancy over a whole frame, then the exact number of non-redundant samples per frame is LM/C_s where M is the number of lines per frame, but the number of nonredundant samples per line is a random variable whose expected value must be determined.

In Chapter II, we found that the run length had an exponential distribution $p(w) = \lambda e^{-\lambda w}$. The expected value of w yields the average sample compression ratio C_s. Hence

$$E(w) = C_s = \int_0^\infty w \lambda e^{-\lambda w} dw = 1/\lambda.$$

or

$$\lambda = 1/C_s .$$

Since the video line has a finite length L, the number N of runs is a r.v. having a Poisson distribution with mean $\lambda L = L/C_s$.

* In practice a value slightly smaller than C_s is chosen, to avoid frequent buffer overflows.

Indeed, referring to Figure 2.1, we form the following r.v.

$$z_N = w_1 + w_2 + \dots + w_N$$

whose probability density is given by

$$f_N(z) = \frac{\lambda^N}{(N-1)!} z^{N-1} e^{-\lambda z} \;;\; N = i, 2, \dots\dots ;\; z > 0$$

as can easily be shown by mathematical induction. The probability distribution $F_N(z)$ given by

$$F_N(z) = \frac{\lambda^N}{(N-1)!} \int_0^z u^{N-1} e^{-\lambda u} \, du$$

is the gamma distribution of order N and parameter λ.

Now the number N of runs in a line of length L is a positive integer and implies that $z_N < L$ and $z_{N+1} \geqslant L$. Thus

$$P(N) = F_N(L) - F_{N+1}(L)$$

and

$$P(N) = \frac{\lambda^N}{(N-1)!} \int_0^L u^{N-1} e^{-\lambda u} \, du - \frac{\lambda^{N+1}}{N!} \int_0^L u^N e^{-\lambda u} \, du$$

Integrating by parts yields

$$(3.14) \qquad P(N) = e^{-\lambda L} \frac{\lambda^N}{N!} = \frac{e^{-L/C_s}}{N!} (L/C_s)^N$$

which is a Poisson distribution with mean L/C_s.

We can start now calculating the m.s. error due to channel noise. We consider the following random variables

n_s number of affected samples per erroneous nonredundant sample

n_r number of erroneous nonredundant samples given that an error has occurred in a run-length word

e_{RL}: error in sample level due to an error in a run-length word.

The total squared error in one line of data given an erroneous run-length word is

$$e^2_{\text{line}} = n_s n_r e^2_{RL}$$

Since there are L samples in a line, the error per sample is

$$e^2_{\text{sample}} = \frac{n_s n_r e^2_{RL}}{L}$$

We can therefore write'

$$E\left(e^2 \Big/ \begin{matrix}\text{error in}\\ \text{RL word}\end{matrix} \right) = \frac{E(n_s n_r e^2_{RL})}{L}$$

or

$$E(e^2) = \sum_x E\left(e^2 \Big/ \begin{matrix}\text{error in}\\ \text{RL word}\end{matrix}\right) \Pr\left(\begin{matrix}\text{x run-length}\\ \text{words in error}\end{matrix} \right)$$

Assuming that the r.v. n_s, n_r and e are uncorrelated, and that no more than one run-length word is in error in any given line, we may write

$$(3.15) \qquad E(e^2) = \frac{E(n_s)\ E(n_r)\ E(e^2_{RL})}{L} \Pr\left(\begin{matrix}\text{one run-length}\\ \text{word in error}\end{matrix} \right)$$

We compute the first factor of (3.15).The actual sample displacement due to an error in the j^{th} bit of a run-length word is 2^{j-1} samples. The number of affected samples out of the n total samples in a run varies with the significance of the bit in error. To obtain the average value of this quantity, we note that when $2^{j-1} \geqslant n$, all samples in that run are affected and $n_s = n$. When $2^{j-1} < n$, we have $n = 2^{j-1}$. The expected value of n_s, given the run-length n, is therefore

$$(3.16) \qquad E(n_s / n) = \sum_{n_s} P(n_s) n_s$$

where n_s varies from 1 to 2^r and $P(n_s)$ denotes the probability of n_s or, equivalently, the probability that a given bit in the r-bit run length word is in error, given that there is an error. We can assume that the error is equally probable in any bit. Thus

$$P(n_s) = 1/r$$

Defining

$$w = [1 + \log_2 n]$$

where $[x]$ denotes the integral part of x, Equation (3.16) can then be written as

$$E(n_s/n) = \frac{1}{r} \left[\sum_{j=1}^{w} 2^{j-1} + \sum_{w+1}^{r} n \right]$$

But we have

$$(3.17) \qquad E(n_s) = \sum_{n=1}^{2^r - 1} E(n_s / n)\, p(n)$$

where p(n) is the probability density of the run length given by (2.7)

$$p(n) = \left[\frac{1}{C_s - 1} \right] \left[\frac{C_s - 1}{C_s} \right]^n$$

Thus we can write

$$E(n_s) = \sum_{n=1}^{2^r - 1} \frac{1}{r} \left[\sum_{j=1}^{w} 2^{j-1} + \sum_{j=w+1}^{r} n \right] \left[\frac{1}{C_s - 1}\right] \left[\frac{C_s - 1}{C_s} \right]^n$$

or

$$(3.18) \quad E(n_s) = \frac{1}{r}\sum_{n=1}^{2^r-1}\left[\sum_{j=1}^{w}2^{j-1} + nr - wn\right]\left(\frac{1}{C_s-1}\right)\left(\frac{C_s-1}{C_s}\right)^n$$

Belver and Hoffman (43) have established the following equation for $E(n_s)$

$$(3.19) \quad E(n_s) = \frac{2^n-1}{r} + 1 - \frac{w}{r}$$

They obtained (3.19) by replacing the actual run length with the average run length C_s. This is a valid procedure only when C_s is a fixed quantity per line of data. This can happen only when averaging has been performed over one line only. If, however, C_s has been obtainee by storing several lines or a whole frame, the compression ratio per line is a r.v. and (3.18) must then be used to obtain $E(n_s)$.

To obtain $E(n_r)$, we note that the average number of nonredundant samples per line is given by L/C_s. Let N be the Poisson distributed r.v. representing the actual number of nonredundant samples in a line. Now, according to the model described at the beginning of this section, the r.v. n_r depends of the position of the run length word in error. Thus an error occurring in the last run length word (i.e., in the $(N-1)^{th}$ word) will cause one erroneous nonredundant sample in the line. An error occurring in the $(N-2)^{th}$ word results in two erroneous nonredundant samples. Similarly, the number of affected nonredundant samples due to an error in the first run length word is (N-1). The probability of an erroneous nonredundant sample, given that there is a run length word error, is 1/N (the probability of error is assumed to be uniformly distributed over the N words). Thus

$$(3.20) \qquad E(n_r / N) = \frac{1}{N} \sum_{i=1}^{N-1} i = \frac{N-1}{2}$$

But

$$(3.21) \qquad E(n_r) = \sum_{N} E(n_r / N)\, P(N)$$

where P(N) is given by (3.14) and N varies from $L/(2^r-1)$ to L, thus ensuring that there is a minimum of $L/(2^r-1)$ runs in a line, or equivalently, a maximum run length equal to 2^r-1.

Equations (3.20) and (3.21) yield

$$(3.22) \qquad E(n_r) = \sum_{N=L/2^r-1}^{L} \left(\frac{N-1}{2} \left(\frac{L}{C_s}\right)^N \frac{e^{-C_s}}{N\,!} \right)$$

which can be simplified in the following way.

$$E(n_r) = \frac{e^{-L/C_s}}{2} \sum_{N} \frac{N-1}{N!} \left(\frac{L}{C_s}\right)^N$$

$$= \frac{e^{-L/C_s}}{2} \sum_{N} \left[\frac{N(L/C_s)^N}{N!} - \frac{(L/C_s)^N}{N!} \right]$$

$$= \frac{e^{-L/C_s}}{2} \sum_{N} \left[\frac{(L/C_s)\ (L/C_s)^{N-1}}{(N-1)!} - \frac{(L/C_s)^N}{N!} \right]$$

For large N we can write

$$\lim_{N\to\infty} E(n_r) = \frac{e^{-L/C_s}}{2} \left[\frac{L}{C_s} e^{L/C_s} - e^{L/C_s} \right]$$

$$= \frac{L/C_s - 1}{2}$$

and

$$(3.23) \qquad \frac{E(n_r)}{E(N)} = \frac{E(n_r)}{L/C_s} = \frac{1 - C_s/L}{2}$$

Thus, if N is Poisson distributed and L is large, the actual number N of nonredundant samples can be replaced by the average number of nonredundant samples L/C_s. Indeed, in this case $E(n_r)/E(N)$ is readily obtained by the following equation

$$E(n_r)/E(N) = \frac{1}{L/C_s}\left[\frac{C_s}{L}\sum_{i=1}^{L/C_s-1} i\right] = \frac{1 - C_s/L}{2}$$

which is identical to (3.23).

The mean square error due to a run length word error is

$$E\left(e_{RL}^2 \,/\, \begin{matrix}\text{sample is}\\ \text{in error}\end{matrix}\right) = E\left\{(x_1-x_2)^2\right\} .$$

where x_1-x_2 is the difference between two adjacent nonredundant samples. Then

$$E\left\{(x_1-x_2)^2\right\} = E(x_1^2) - 2E(x_1x_2) + E(x_2^2).$$

$E(x_1x_2)$ is a function of the correlation between samples, and depends, therefore, on the source statistics. If the statistics are not known, an upper bound to (2.1) is obtained by assuming that the random variables x_1 and x_2 are uncorrelated *. Under this assumption, we have

$$\begin{aligned} E\left\{(x_1-x_2)^2\right\} &= E(x_1^2) - 2E(x_1)E(x_2) + E(x_2^2) \\ &= 2\,E(x_1^2) - E^2(x_1) \quad . \end{aligned}$$

The r.v. x_1 can have any discrete values $j/2^k$ where $0 \leq j \leq 2^k-1$. Assuming that x_1 is uniformly distributed in the interval (0,1) we may write

$$P_j = 1/2^k$$

Then

$$E^2(x_1) = \left[\sum_{j=0}^{2^k-1} P_j \frac{j}{2^k} \right]^2$$

$$= \left[\frac{1}{2^{2k}} \sum_{j=0}^{2^k-1} j \right]^2 = \left[\frac{2^k \, (2^k - 1)}{2^k \, 2} \right]^2$$

$$= \frac{(2^k - 1)^2}{2^{2(k+1)}}$$

We also have

$$E(x_1^2) = \sum_{j=0}^{2^k-1} P_j \, j^2/2^{2k} = \frac{1}{2^{3k}} \sum_{j=0}^{2^k-1} j^2$$

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

Therefore,

$$E(x_1^2) - E^2(x_1) = \frac{1}{12} \frac{2^{2k} - 1}{2^{2k}}$$

and

$$E\left\{(x_1 - x_2)^2\right\} = \frac{1}{6} \frac{2^{2k} - 1}{2^{2k}} \tag{3.24}$$

Suppose now that successive nonredundant samples are correlated and let ρ be their correlation coefficient

$$\rho = \frac{\text{cov } (x_1 x_2)}{\sigma_{x_1} \sigma_{x_2}} = \frac{E(x_1 x_2) - E(x_1)^2}{\sigma_{x_1} \sigma_{x_2}}$$

We have

$$\sigma^2_{x_1} = \sigma^2_{x_2} = E(x_1^2) - E^2(x_1) = \frac{1}{12} \frac{2^{2k} - 1}{2^{2k}}$$

and

$$E(x_1x_2) = \rho\sigma^2_{x_1} + E(x_1)^2$$

$$E\left\{(x_1-x_2)^2\right\} = E(x_1^2) - 2E(x_1x_2) + E(x_2^2)$$

$$= 2\ E(x_1^2) - E(x_1x_2)$$

Hence,

$$(3.25) \qquad E\left\{(x_1-x_2)^2\right\} = 2(1-\rho)\,\sigma^2_{x_1} = \frac{1-\rho}{6}\left[\frac{2^{2k}-1}{2^{2k}}\right]$$

The last factor of (3.15) is the probability P(RLE) that one run length word is in error in a line. If the bit error probability is P_B,

$$(3.26) \qquad P(RLE) = \frac{rL}{C_s}P_B\left(1 - \frac{rL}{C_s}P_B\right)^{\frac{rL}{C_s}} \cong \frac{rLP_B}{C_s}$$

for small values of P_B ($P_B < 10^{-3}$).

Note that the expected number of run length words in error is given by

$$E(N) = \sum_N N\binom{L/C_s}{N}(rP_B)^N(1-rP_B)^{L/C_s-N}$$

$$= \frac{rLP_B}{C_s}$$

Hence, for values of $P_B < 10^{-3}$, the expected number of RL words in error is smaller than 1; thus it is reasonable to assume no more than one error per line for most practical channels.

Substituting (3.26), (3.24) and (3.23) into (3.15) yields the m.s. error caused by channel noise in the case of run length encoding

$$(3.27) \qquad e^2 = \left[\frac{rLP_B}{12C_s}\right]\left[\frac{2^{2k}-1}{2^{2k}}\right]\left[\frac{2^w-1}{nr} + 1 - \frac{w}{r}\right]\left[\frac{1}{C_s-1}\right]\left[\frac{C_s-1}{C_s}\right]^n$$

B. Single Address Word Encoding

As explained in Chapter I (Section 1.1), the address word consists of L bits in this case, a zero in the ith bit of the address word indicating that the ith sample is redundant while a one indicates that the sample is nonredundant.

For example, suppose that the following samples have been transmitted (the number below indicates the value of the sample level)

2 3 4 3 1

together with the following address word

0 0 0 1 0 1 0 0 1 1 0 0 0 1

At the receiver, a 0 is interpreted as a repeat indication and a 1 as a call for a new sample level.

Thus the restored data will take the following form

2 2 2 2 3 3 4 4 4 3 1 1 1 1

0 0 0 1 0 1 0 0 1 1 0 0 0 1

Moreover, we suppose that the receiver has some decoding ability when there is a disagreement between the received binary sequence and the number of received sample levels. The address word must be modified so as to use all sample levels; the following decoding rule is then applied by the receiver

1. If there are excess 1's, the terminal excess 1's are changed to 0's.

2. If there are excess 0's, the terminal excess 0's are changed to 1's.

In this manner, there are no excess samples which cannot be inserted in the flow of data, or empty slots which are left without data.

It is evident from this example that, as in the run length encoding technique, a single error in the address word affects not only the sample that it represents , but also the remainder of the line. This error propagation can, therefore, be analyzed in the same manner as for run length encoding. Thus we have

$$E(e^2_{SA} / \begin{matrix}\text{given an error}\\ \text{in address word}\end{matrix}) = \frac{1}{6}\left(\frac{2^{2k}-1}{2^{2k}}\right)$$

$$E(n_r) = \frac{1 + C_s/L}{2}$$

The average number of errors in the single address word is

$$E(\begin{matrix}\text{address words}\\ \text{in error}\end{matrix}) = \sum_{i=1}^{L} i \binom{L}{i} P_B^i \ (1-P_B)^{L-i} = LP_B$$

To calculate the expected number of affected redundant samples per redundant sample, we observe that the maximum amount of affected samples in a run of length n is n, i.e., all samples are erroneous; this happens if a one marking the beginning of the run has been changed to zero. In general there will be 1, 2, ..., n-1 erroneous samples if the (n-1)th, ..., second, first zero respectively, are interpreted as one. This is illustrated by Figure 3.4 for a received sequence of three nonredundant samples of values 4, 3, 2. We look at the second run whose length is 8 and which consists of samples of amplitude 3.

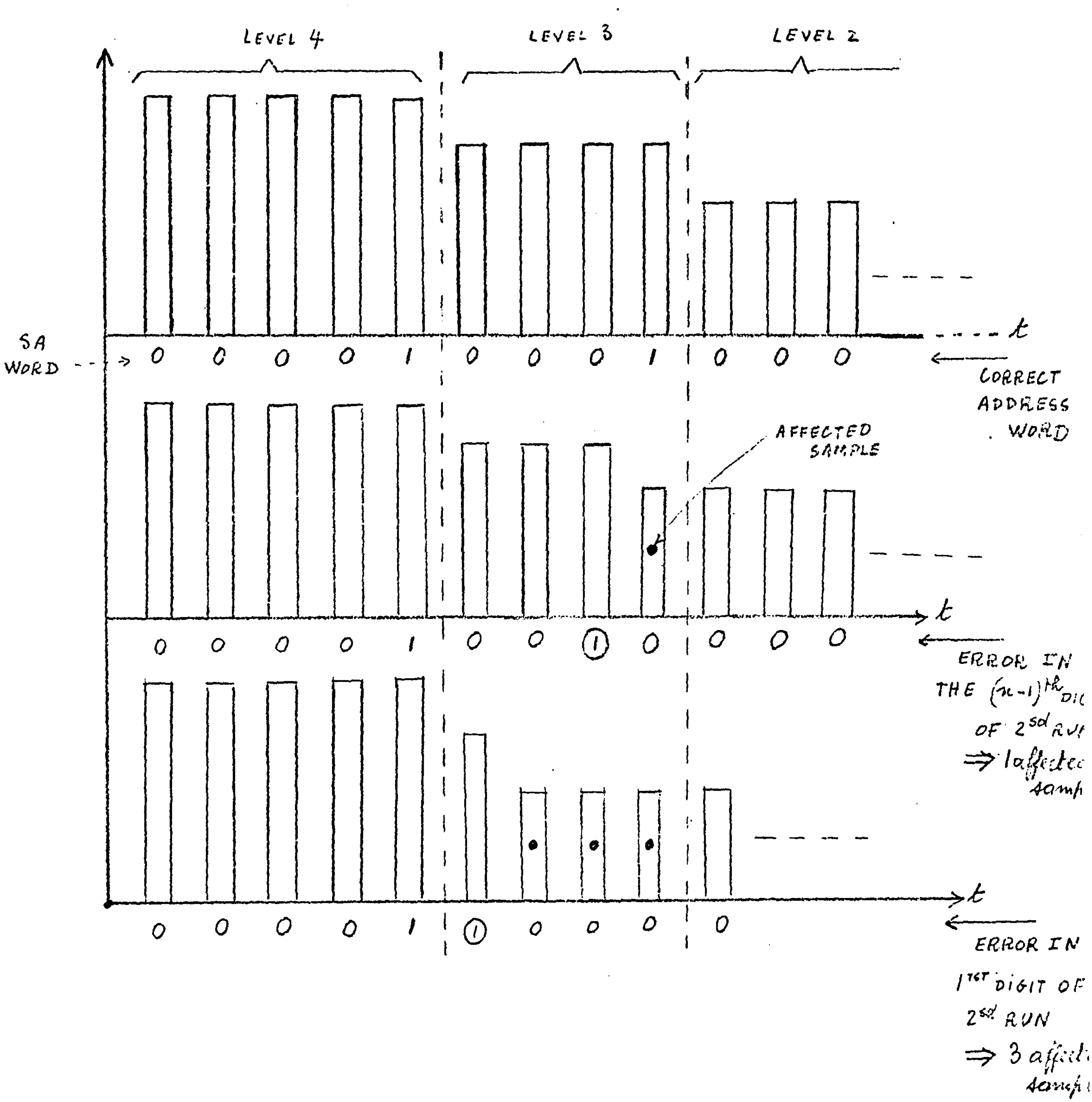


Figure 3.4 EFFECT OF BIT ERRORS IN ADDRESS WORD

Given that there is an error in the address word, it is reasonable to assume that the location of the bit in error is uniformly distributed between the n locations. Hence,

$$\Pr(j^{th} \text{ bit in error}) = 1/n$$

from which we obtain

$$E\left(n_s \,/\, \begin{matrix}\text{error in address word}\\ \text{and run length } n\end{matrix}\right) = \frac{1}{n} \sum_{j=1}^{n} j = \frac{n+1}{2}$$

and

$$E\left(n_s \,/\, \begin{matrix}\text{error in}\\ \text{address word}\end{matrix}\right) = \sum_{n=1}^{L} \frac{n+1}{2}\, p(n)$$

where p(n) is the probability of the run length.

Hence, for large values of L we obtain the following approximation

$$\text{(3.28)} \qquad E\left(n_s \,/\, \begin{matrix}\text{error in}\\ \text{address word}\end{matrix}\right) \cong (C_s+1)/2.$$

Since the m.s. error due to channel noise can be expressed as

$$E(e_{SA}^2) = \sum_{j=1}^{L} \frac{E(n_r)}{L/C_s} \frac{E(n_s)}{C_s} E\left(e_{SA}^2 \,/\, \begin{matrix}\text{given error}\\ \text{in address}\end{matrix}\right) \Pr\left(\begin{matrix}\gamma \text{ address wds}\\ \text{in error}\end{matrix}\right)$$

we obtain by substitution

$$\text{(3.29)} \qquad E(e_{SA}^2) = \frac{LP_B}{6} \cdot \frac{2^{2k}-1}{2^{2k}} \cdot \frac{1+C_s/L}{2} \cdot \frac{C_s+1}{2C_s}$$

C. Position Encoding

We have seen in Chapter I that this encoding method is less efficient regarding the compression ratio than run-length encoding. However, position encoding is much less sensitive to noise than any other addressing scheme and, in many practical applications, might need little or no coding to yield an acceptable m.s. error. Also, its performance does not depend heavily on the length of the line, as is the case with run-length encoding or single-bit encoding. In the case of picturial data, this means that several frames could be stored and some "replenishment" technique be performed without serious picture degradation. However, this addressing scheme has not been studied in the literature, due perhaps to the fact that it is less efficient than others for deep space telemetry. In the following, we will compute the m.s. error due to the effect of noise on the timing word.

Figure 3.5 shows a typical pattern of nonredundant (i.e., transmitted) samples. For convenience, the samples are shown in their unquantized form. Each nonredundant sample is tagged by an h-bit word, where $h = \log_2 L$, describing the position of the sample along the line. There are $\beta = L/C_s$ nonredundant samples in a line and thus β position words; as in the preceding case, the sample level is described by a k-bit word. We suppose that the receiver performs in the following way (see Figure 3.6). It transfers the synchronous information stored in buffer A to buffer B, placing each sample in buffer B according to their indicated position.(In this way the receiver has reconstructed -assuming error-free transmission -- the asynchronous data as it appeared at the output of the compressor) We also assume that the box called "address decoder" has a

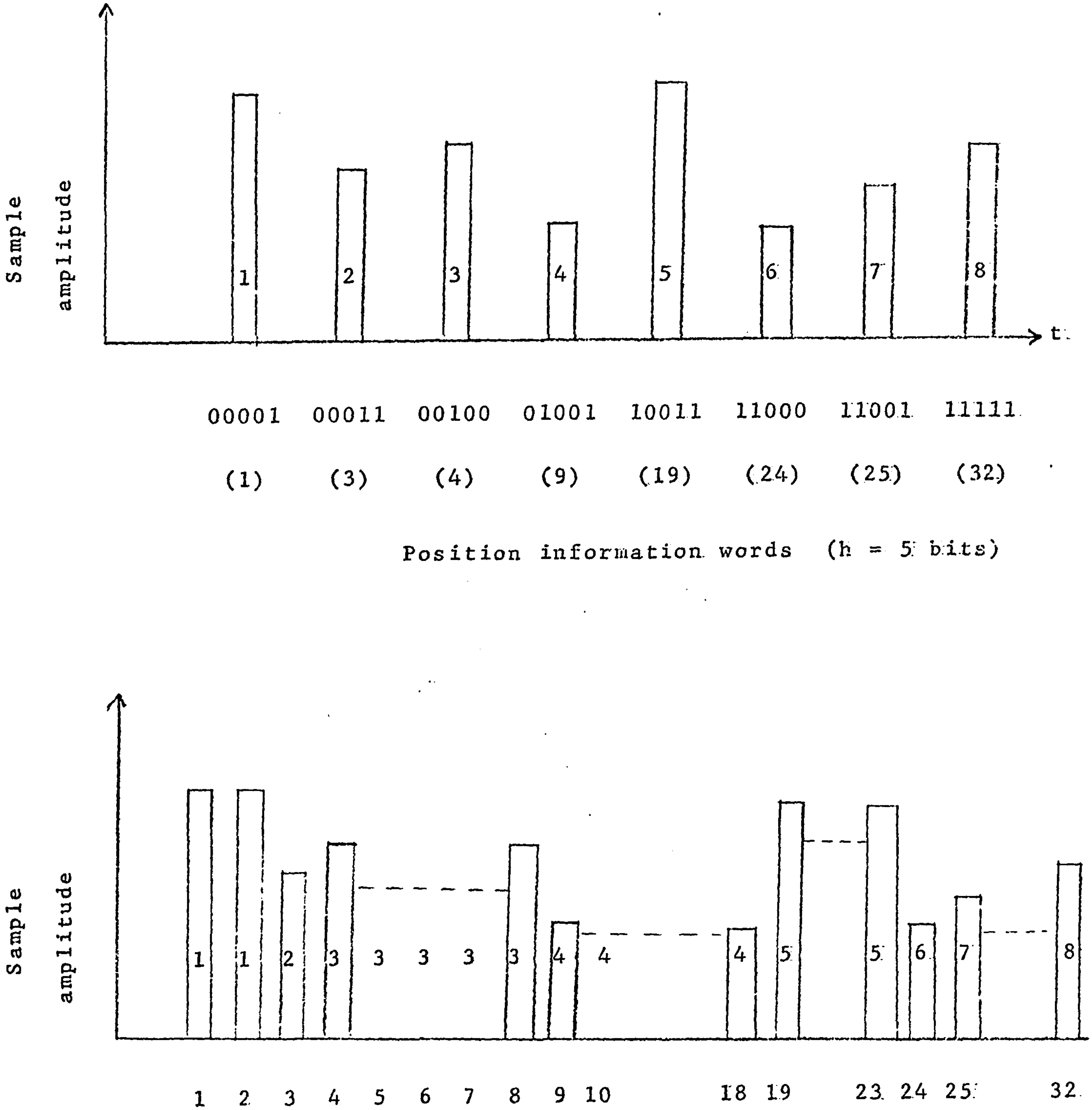


Figure 3.5 POSITION ENCODING

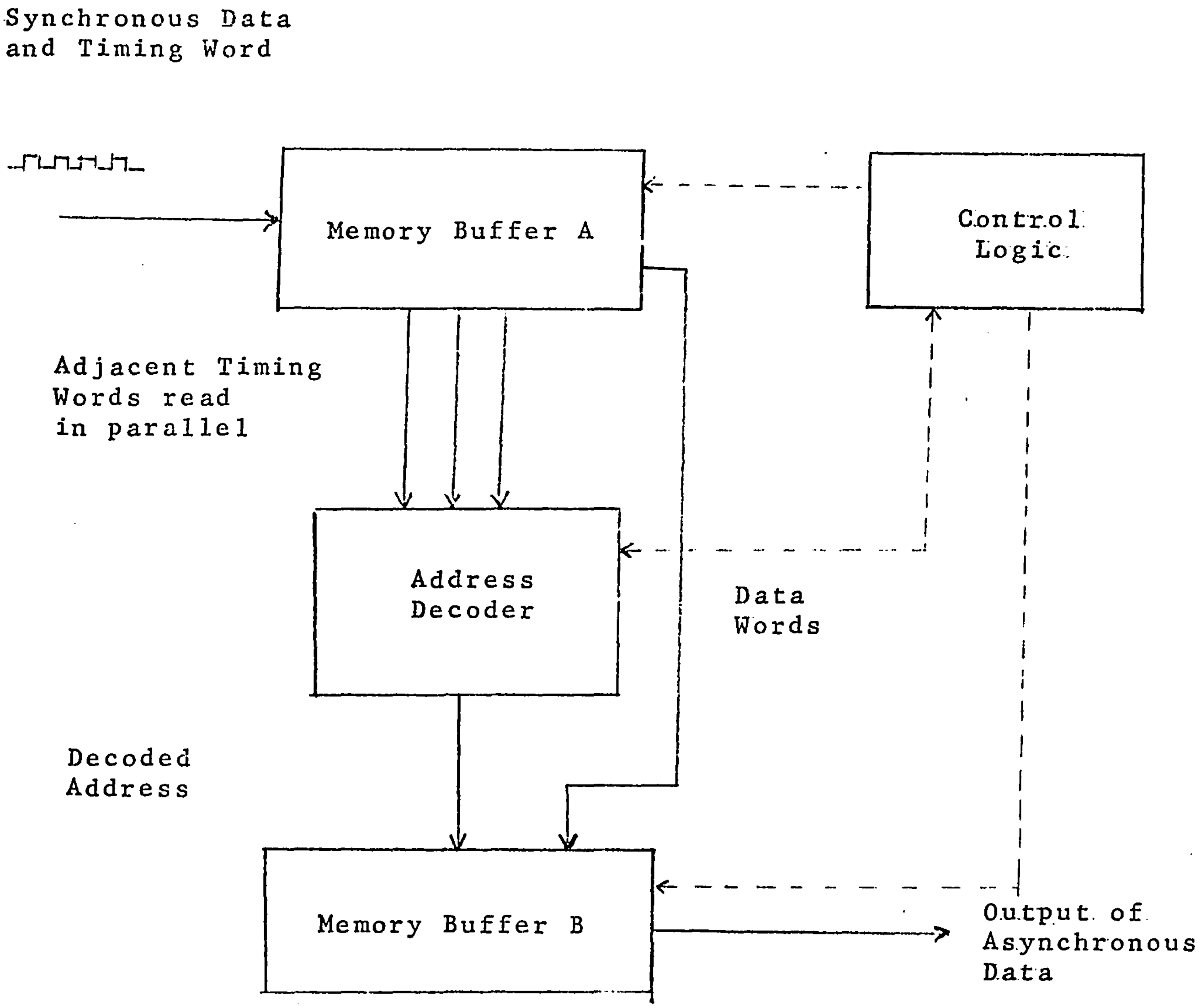


Figure 3.6 DECODING OF POSITION WORD

certain error-correcting ability; it reads the three consecutive timing words at a time, say R_i, R_{i+1}, R_{i+2} pertaining to samples S_i, S_{i+1}, S_{i+2} respectively. Thus the following three cases arise

Case 1. If $R_i < R_{i+1} < R_{i+2}$, the address decoder assumes that the words are correct and thus allows the samples S_i, S_{i+1}, S_{i+2} to be positionned accordingly.

Case 2. If $R_{i+1} < R_i < R_{i+2}$, the address decoder performs again as above (since it is impossible for the decoder to know which word is in error, R_i or R_{i+1}).

Case 3. If $R_{i+1} < R_{i+2} < R_i$, the decoder assumes R_i is in error (since this inequality can be true only if both consecutive timing words R_{i+1} and R_{i+2} are in error, an event which has a very small probability of occurrence). The decoder then assigns to S_i a position R_i' such that $R_{i-1} < R_i' < R_{i+1}$. One way of implementing this inequality is to place R_i' half-way between R_{i-1} and R_{i+1}. Thus, with this method, the average error displacement of a given sample does not exceed $2C_s$.

The expected squared error due to a position word error is given by

$$(3.30) \qquad E(e_p^2) = \frac{E(n_s/\text{error})}{C_s} \; E(e_p/\text{error}) \; \Pr\left(\begin{matrix}\text{a position word}\\ \text{is in error}\end{matrix}\right)$$

where

$$E(n_s\ /\text{error}) = E\left(\begin{matrix}\text{number of erroneous samples due}\\ \text{to an error in the position word}\end{matrix}\right)$$

$$E(e_p^2/\text{error}) = E\left\{(x_1-x_2)^2\right\} = \frac{1}{6} \cdot \frac{2^{2k} - 1}{2^{2k}}$$

$$\Pr\left(\begin{matrix}\text{position word}\\ \text{in error}\end{matrix}\right) = 1 - (1 - P_B)^{Lh/C_s} \cong \frac{Lh}{C_s} P_B \text{ , } \quad P_B \leqslant 10^{-4}$$

where h is the length of the address word.

Note that an error in a position word does not cause any propagation of erroneous samples as in the case of run-length encoding, but is limited to a fixed number of samples.

The average number of affected samples given an error must account for the three possibilities listed above. Hence we must have

$$(3.31) \qquad E(n_s/\text{error}) = \frac{1}{n} \sum_{i=1}^{3} \Pr(\text{case } i)\ E(n_{s_i}/\text{error})$$

where

$\Pr(\text{case } i)$ = Probability of being in state i

$E(n_{s_i}/\text{error})$ = Expected displacement for case i

on a per sample basis

Now the displacement due to an error on the j^{th} bit of the position word is 2^{j-1}. Let n represent the run length and $w = [\,1 + \log_2 n\,]$ be the number of digits in the position word such that an error occurring in any of the w digits would cause a displacement less than n, thereby fulfilling the requirements of state 1. Then

$$E\left(n_{s_1}/\begin{matrix}\text{given an error}\\ \text{and given n}\end{matrix}\right) = \frac{1}{n} \sum_{j=1}^{w} P_j\ 2^{j-1} = \frac{2^w - 1}{nw}$$

Since we know p(n), we obtain

$$E(n_{s_1}/\text{error}) = \sum_{n=1}^{2^h - 1} \frac{2^w - 1}{nw}\ p(n)$$

or

(3.32) $$E(n_{s_1}/\text{error}) = \sum_{n=1}^{2^h-1} \left[\frac{2^w - 1}{nw} \; \frac{1}{C_s - 1} \; \left(\frac{C_s - 1}{C_s}\right)^n \right]$$

where $w = \left[1 + \log_2 n\right]$.

Since an error is equally likely in each of the h bits of a position word, state 1 will occur with probability

$$\Pr(\text{case } 1)_n = w/h \quad \text{for each } n.$$

Case 2 occurs when $j = w + 1$, yielding

$$E(n_{s_2}/{}^{\text{given an error}}_{\text{and given n}}) = 2^w/n$$

and

(3.33) $$\begin{aligned} E(n_{s_2}/\text{error}) &= \frac{1}{n} \sum_{n=1}^{2^h-1} 2^w \, p(n) \\ &= \frac{1}{n} \sum_{n=1}^{2^h-1} 2^{\left[\log_2 n + 1\right]} \; \frac{1}{C_s - 1} \; \left(\frac{C_s - 1}{C_s}\right)^n \end{aligned}$$

The probability of occurrence of state 2 is

$$\Pr(\text{case } 2) = 1/h \quad \text{independent of } n.$$

Finally, in case 3, the error is detected and corrected by the receiver, which assigns to the erroneous position the value

$$\frac{\text{Position}(n-1) + \text{Position}(n+1)}{2} .$$

Assuming that the r.v. representing the correct value of position n is uniformly distributed from Position(n-1) + 1 to Position(n+1) + 1 , we obtain

$$E(n_{s3}/{\text{given an error} \atop \text{and given n}}) = \frac{2}{n(2n-1)} \sum_{j=1}^{n-1} j = \frac{n-1}{2n-1}$$

and

$$E(n_{s3}/\text{error}) = \sum_{n=1}^{2^h-1} \frac{n-1}{2n-1} \; \frac{1}{C_s-1} \; \left(\frac{C-1}{C_s}\right)^n$$

The probability of being in state 3 is

$$\Pr(\text{case } 3)_n = 1 - \frac{w+1}{h} \quad \text{for} \quad n = 1, 2, \ldots\ldots, 2^h-1$$

and (3.31) can now be expressed in the following manner

$$(3.34) \quad E(n_s/\text{error}) = \sum_{n=1}^{2^h-1} \left\{ \frac{1}{C_s-1} \left(\frac{C_s-1}{C_s}\right)^n \left[\frac{2^{w+1}-1}{hn} + \left(1-\frac{w+1}{h}\right)\left(\frac{n-1}{2n-1}\right) \right] \right\}$$

Combining (3.30) and (3.34), we obtain the m.s. error due to timing error for position encoding systems.

$$(3.35) \quad E(e_p^2) = \frac{LP_B}{6C_s} \cdot \frac{2^{2k}-1}{2^{2k}} \sum_n \frac{1}{C_s-1} \left(\frac{C_s-1}{C_s}\right)^n \left[\frac{2^{w+1}-1}{n} + \frac{(h-w+1)(n-1)}{2n-1} \right]$$

The total r.m.s. error in the reconstructed data is the sum of the four independent errors. We have considered in this chapter quantization, aperture, level word and timing word errors due to channel noise.

Combining the expressions found for each of these errors we obtain the following result for the three addressing schemes described in this study.

1. Run length encoding

$$(3.36) \quad e_{RMS} = \left\{ \frac{1}{12.2^{2k}} + \frac{m(m+1)(C_s-1)}{3C_s(2^k-1)^2} + \frac{P_B}{3}\left(\frac{2^{2k}-1}{2^{2k}}\right) + \frac{rLP_B}{6\ C} \; \frac{2^{2k}-1}{2^{2k}} \; \frac{1-C_s/L}{2} \left[\sum_{n=1}^{2^r-1} \left(\frac{2^w-1}{nr} + 1 - \frac{w}{r}\right)\left(\frac{1}{C_s-1}\right)\left(\frac{C_s-1}{C_s}\right)^n \right] \right\}^{\frac{1}{2}}$$

2. Single address word encoding

$$(3.37)\quad e_{RMS} = \left\{ \frac{1}{12.2^{2k}} + \frac{m(m+1)(C_s-1)}{3C_s(2^k-1)^2} + \frac{P_B}{3}.\frac{2^{2k}-1}{2^{2k}} + \frac{LP_B}{6}.\frac{2^{2k}-1}{2^{2k}}.\frac{1+C_s/L}{2}.(\frac{C_s-1}{2C_s}) \right\}^{\frac{1}{2}}$$

3. Position word encoding

$$(3.38)\quad e_{RMS} = \left\{ \frac{1}{12.2^{2k}} + \frac{m(m+1)(C_s-1)}{3C_s(2^k-1)} + \frac{P_B}{3}.\frac{2^{2k}-1}{2^{2k}} + \frac{LP_B}{6C_s}.\frac{2^{2k}-1}{2^{2k}} \sum_{n=1}^{2^r-1} \frac{1}{C_s-1}\;\frac{C_s-1}{C_s}\left[\frac{2^{w+1}-1}{n} + \frac{(h-w+1)(n-1)}{2n-1}\right] \right\}^{\frac{1}{2}}$$

Each of the above equations is expressed in terms of the ratio of r.m.s. error to full scale amplitude. The r.m.s. error in a non-compressed signal is used to evaluate the effect of the compression scheme on the reconstructed data. The r.m.s. error for a PCM signal is given by (see (3.2))

$$e_{RMS} = \left\{ \frac{1}{12.2^{2k}} + \frac{P_B}{3}.\frac{2^{2k}-1}{2^{2k}} \right\}^{\frac{1}{2}}$$

Equations (3.36), (3.37) and (3.38) have been calculated for certain practical values of the parameters k, h, r and C_s. Thus all the curves have been plotted for k=6 bits, and zero aperture error. The total number of samples per line is taken as 1000 and the run lengths have been restricted to a maximum of 32 samples. Thus r = 5. For the position encoding scheme, the word length is h = $\log_2 1000$ = 10. The size k of the level word yields a r.m.s. quantization error of 0.45 %. This error is not a function of the bit error probability and, therefore the curves of Figures 3.7 and 3.8 illustrate the magnitude of the error versus both sample compression ratio and bit error probability. It is seen that errors in run length encoding and single address encoding systems vary significantly as a function of Cs. Position encoding systems give clearly a better performance than the other techniques (at the expense, however, of a smaller bit compression ratio). It is interesting to note that the r.m.s. error decreases with C_s for the run length encoding technique while it increases with C_s for position encoding.

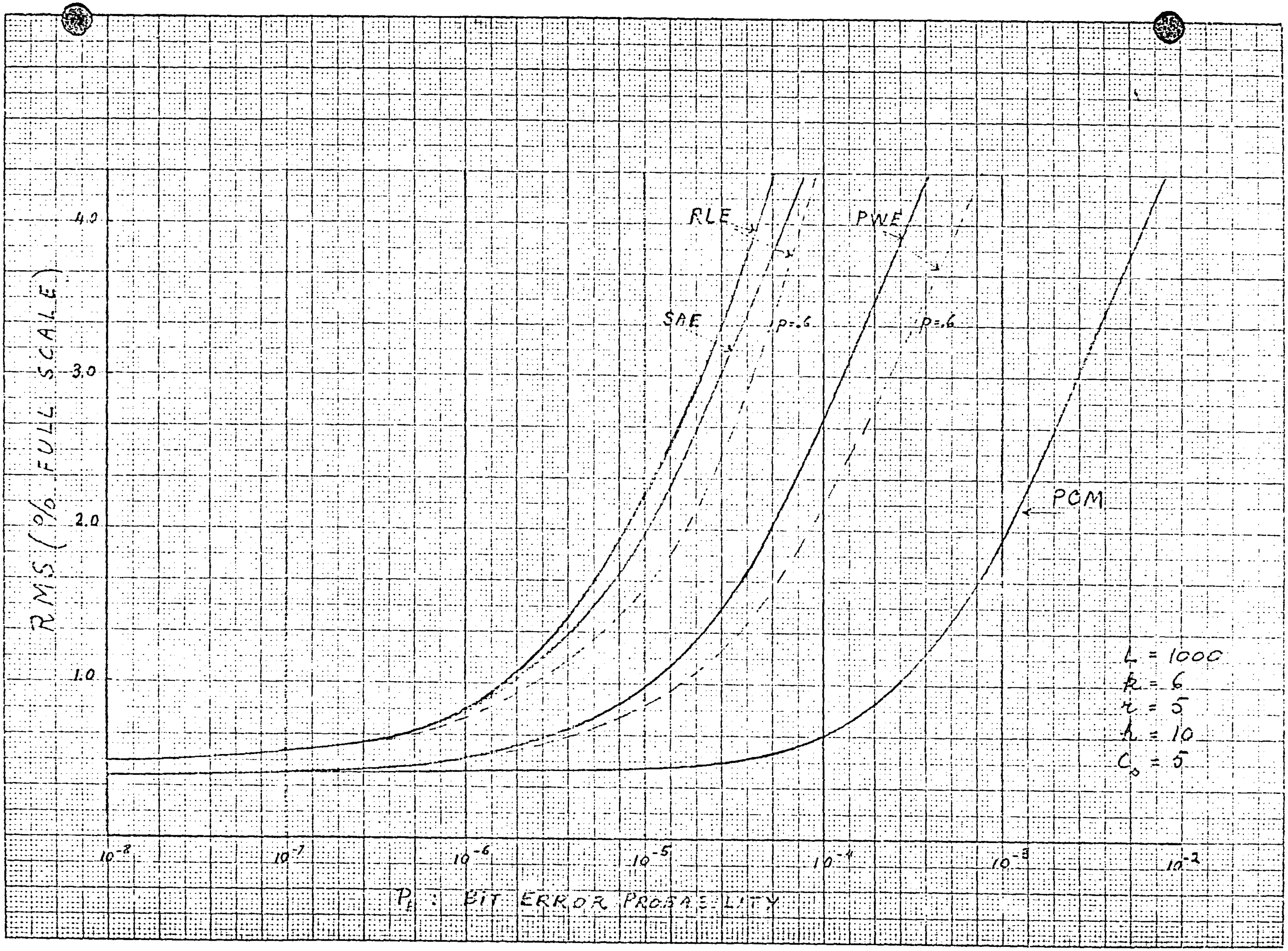

RLE
PWE
SAE
p=.6
p=.6
PCM
4.0
3.0
2.0
1.0
RMS (% FULL SCALE)
10^{-8}
10^{-7}
10^{-6}
10^{-5}
10^{-4}
10^{-3}
10^{-2}
P_E : BIT ERROR PROBABILITY
L = 1000
k = 6
r = 5
h = 10
C_0 = 5

Figure 3.7

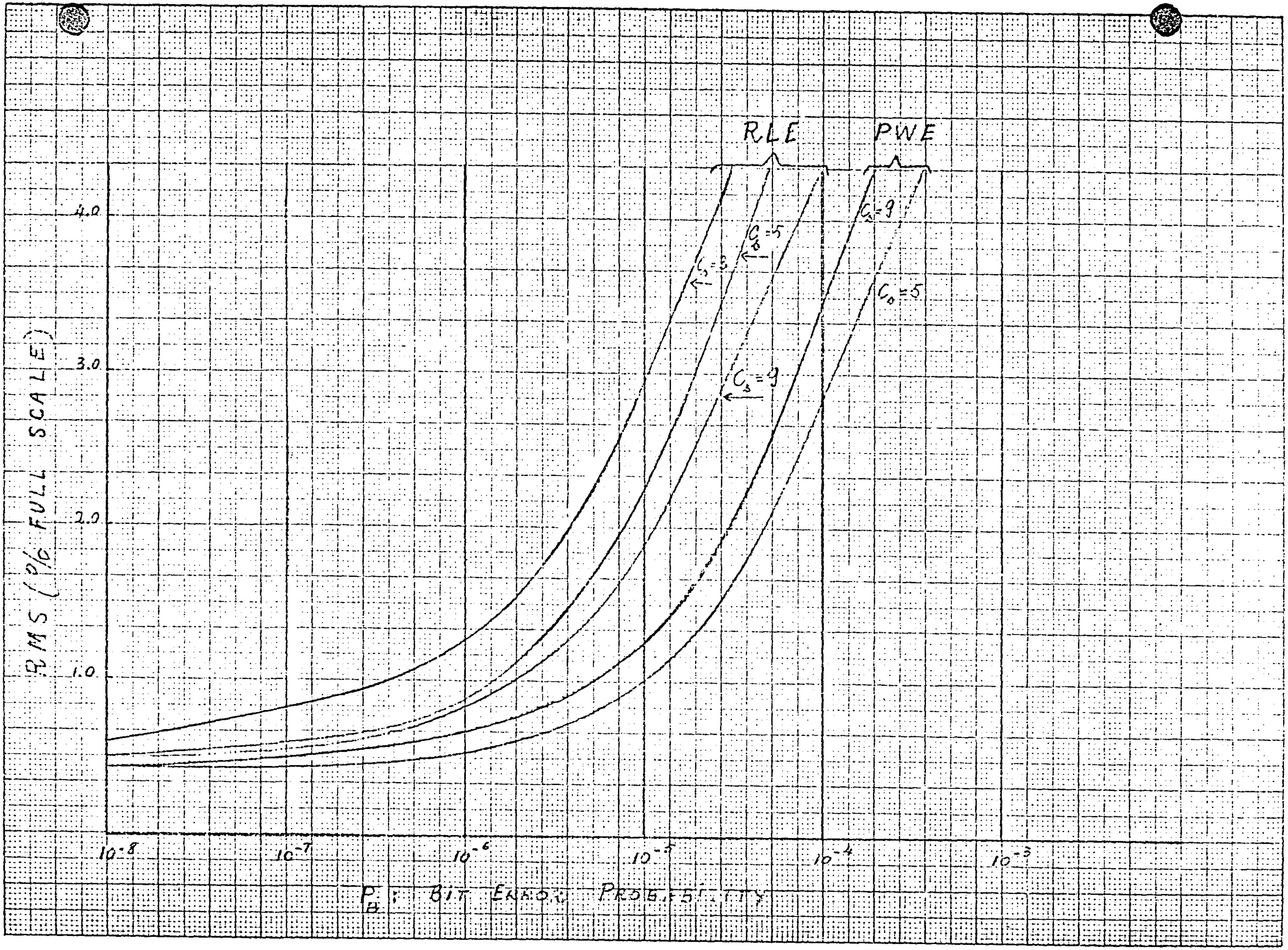

RLE
PWE
$C_s=3$
$C_s=5$
$C_s=9$
$C_s=9$
$C_s=5$
RMS (% FULL SCALE)
4.0
3.0
2.0
1.0
10^{-8}
10^{-7}
10^{-6}
10^{-5}
10^{-4}
10^{-3}
P_B : BIT ERROR PROBABILITY

Figure 3.8

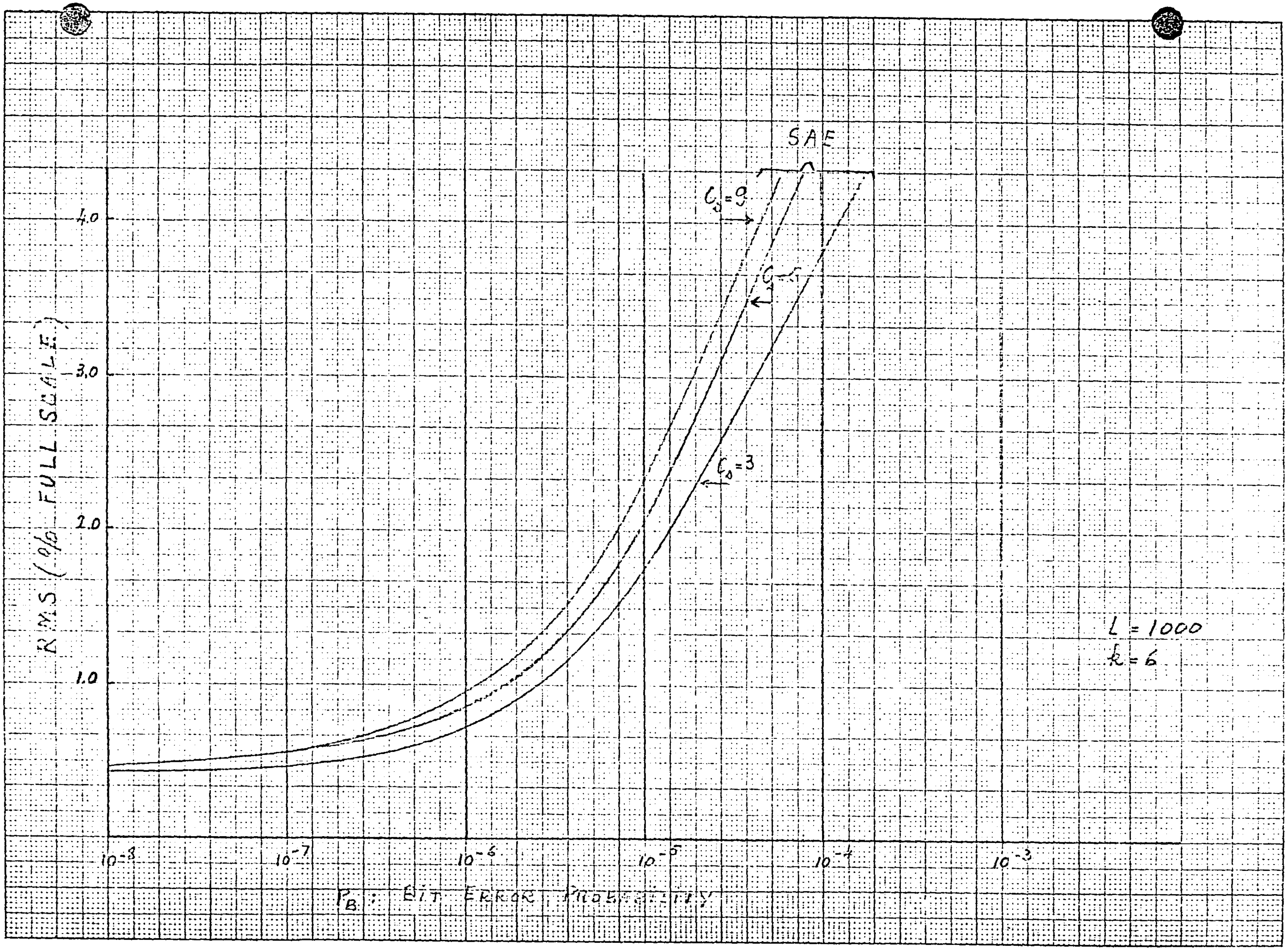

SAE
C_s=9
C_s=5
C_s=3
L = 1000
k = 6
4.0
3.0
2.0
1.0
RMS (% FULL SCALE)
10^{-8}
10^{-7}
10^{-6}
10^{-5}
10^{-4}
10^{-3}
P_B : BIT ERROR PROBABILITY

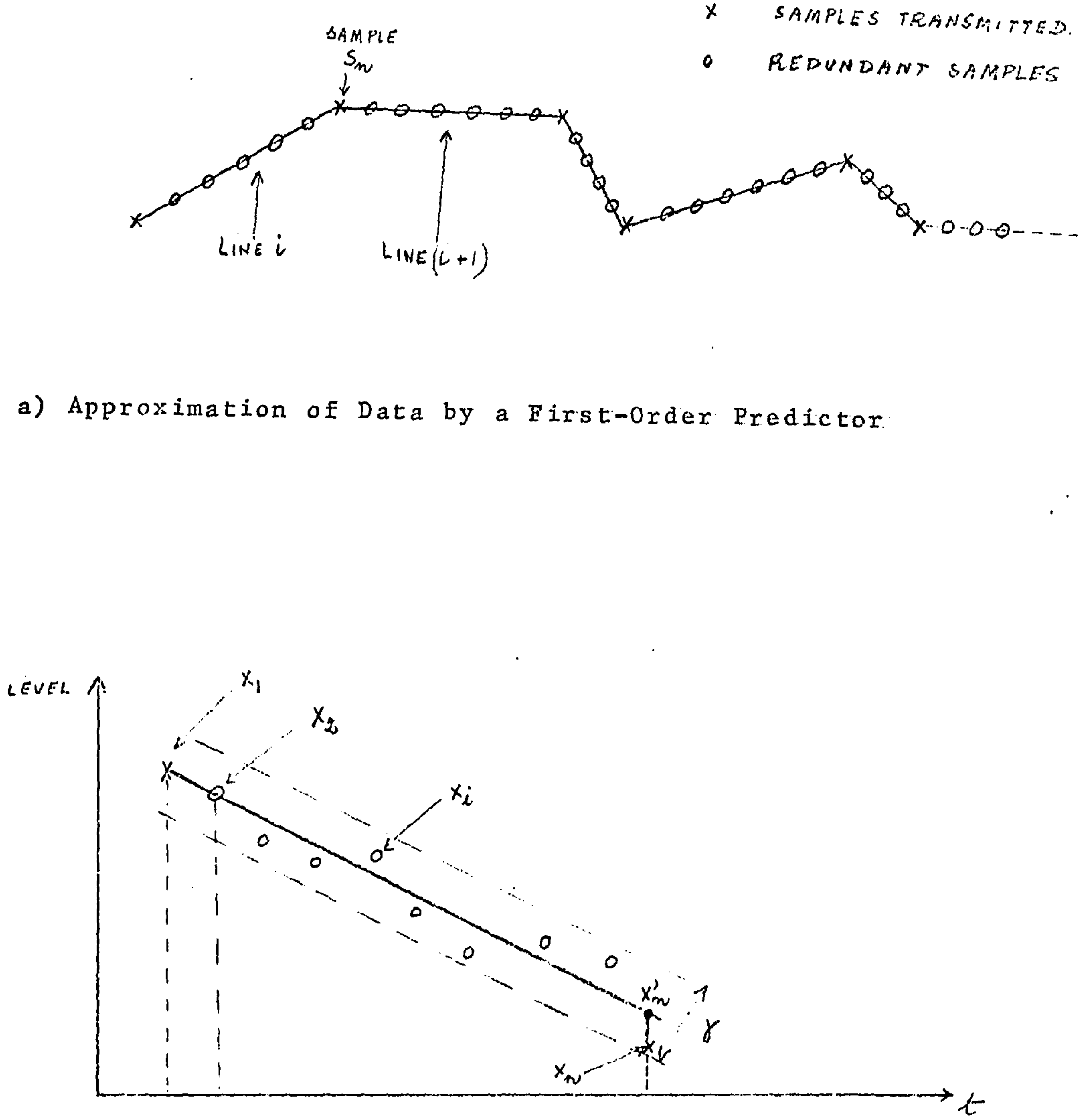
Figure 3.9

CHAPTER IV

EFFECT OF CHANNEL NOISE ON FIRST-ORDER PREDICTOR

4.1 Introduction

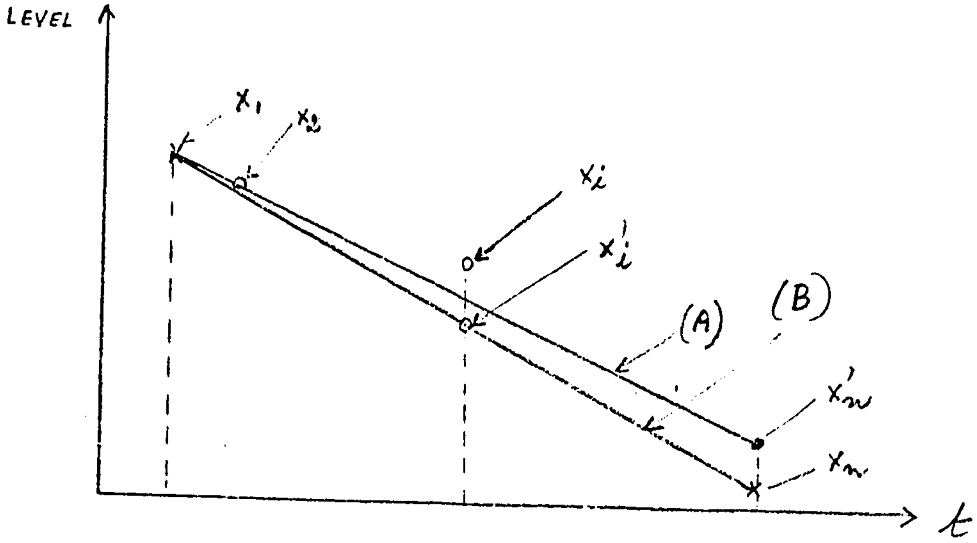
We have seen in Chapter I that a first-order polynomial predictor performs a prediction of the amplitude of a given sample by considering both amplitude and slope obtained from previous samples. In this chapter, we shall use a simple first-order predictor which uses two adjacent samples to predict the amplitude of succeeding samples, and transmits the starting and end points of a straight line. The end point of line j constitutes the starting point of line (j+1). Thus the original analog waveform is approximated by a succession of lines as shown in Figure 4.1. We wish to calculate now the difference between the true value of a sample, say sample X_i, and the interpolated value X_i' obtained at the receiver. The transmitter stores the values of X_1 and X_2 and draws a line passing through these two points. If the amplitude of sample X_3 falls within a distance $\gamma/2$ from this line, then X_3 is considered redundant, and sample X_4 is compared next. The process continues until a sample is found exceeding the given tolerance. Suppose that N samples $X_1, X_2, \ldots, X_N$ fall within the aperture γ (see Figure 4.2). The true value of the i^{th} sample is



a) Approximation of Data by a First-Order Predictor

b) Transmitter Operation

Figure 4.1 FIRST-ORDER PREDICTOR ALGORITHM



(A) Line at transmitter

(B) Line at receiver

Figure 4.2 RECEIVER OPERATION

given by

$$X_i = X_1 - \frac{X_1 - X_n'}{n - 1}(i-1) + \Delta_i$$

where Δ_i = ma is a zero mean discrete r.v. $(m = 0, \pm 1, \ldots)$
$(i = 1, 2, \ldots, n)$

a = quantization step

n = length of the run

The transmitter sends samples X_1 and X_n to the receiver; it is clear that

$$X_n = X_n' + \Delta_n$$

Therefore, at the receiver, the value of sample X is calculated as

$$X_i' = X_1 - \frac{X_1 - X_n}{n-1}(i-1)$$

$$= X_1 - \frac{X_1 - X_n - \Delta_n}{n-1}(i-1)$$

We can assume that the r.v. Δ_i $(i = 1, \ldots, n)$ are independent. Therefore the error in the reconstructed data is

$$e_A = S_i - S_i' = \frac{\Delta[n-i]}{n-1} .$$

Since $E(\Delta)E(i) = 0$, the m.s. error due to an aperture is

$$(4.1) \quad E(e_s^2/n) = \frac{n^2 E(\Delta^2/n) + E(\Delta^2/n)E(i^2)}{(n-1)^2}$$

$$= E(\Delta^2/n)\,\frac{n^2 + E(i^2)}{(n-1)^2}$$

and

(4.2) $$E(e_A^2) = \sum_n E(e_A^2/n)\ p(n)$$

where p(n) is the run length distribution for a first-order predictor. In order to determine p(n), however, we must know the statistical structure of the source.

4.1.1 Run Length Distribution

The determination of the run length distribution of a first-order system requires knowledge of

a) the power density spectrum of the data, and

b) the probability density of the amplitude of the data.

M.Bruce (46) has computed the theoretical compression ratio for a first-order predictor operating on four types of input data; the probability density of the amplitude of the data is Gaussian with a zero mean and the standard deviation is σ . The following amplitude spectra were examined.

L spectrum

$F(w) = k_1$ $\quad (0 \leqslant w \leqslant 0.1\ w_m)$

$F(w) = 0.001\ k_1$ $\quad (0.1\ w_m \leqslant w \leqslant w_m)$

$F(w) = 0$ $\quad$ elsewhere

Exponential spectrum

$F(w) = k_2 \exp(- \frac{5\ w}{w_m})$ $\quad (0 \leqslant w \leqslant w_m)$

$F(w) = 0$ $\quad$ elsewhere

Triangular spectrum

$F(w) = k_3 (1 - \frac{w}{w_m})$ $\quad (0 \leqslant w \leqslant w_m)$

$F(w) = 0$ $\quad$ elsewhere

Rectangular spectrum

$$F(w) = k_4 \qquad (0 \leqslant w \leqslant w_m)$$

$$F(w) = 0 \qquad \text{elsewhere}$$

Figure 4.3 is a plot of the theoretical compression ratio on the four types of input data. The abscissa K is the magnitude of the tolerance. Clearly, there is a trade-off between the amount of compression ratio and the width of the aperture. Moreover, the compression ratio depends on the spectrum of the input data.

If the statistics of the input data are not available, one can still assume a certain data structure; for example, the data could be a Markov process, and we have seen in Chapter II that TV signals are approximately first-order Markov where the highest transition probability is from the present level to the same level. Thus the input data can efficiently be approximated by a succession of straight horizontal lines, which is exactly what a zero order predictor does. But then, as noted by Davisson (47), for a large value of the transition probability p, the performance of a first-order system is inferior to that of the zero-order system. Most interpolating lines are indeed horizontal and the former transmits two samples for each run, while the latter transmits only one (i.e., the start of the line only) per run. On the other hand, for small values of p ($p < 0.5$), the first-order method gives better results than the zero-order one, but the resulting compression ratio is small when PCM encoding is taken as the standard of comparison. However, a first-order method may not be inferior for all kinds of data. In particular,

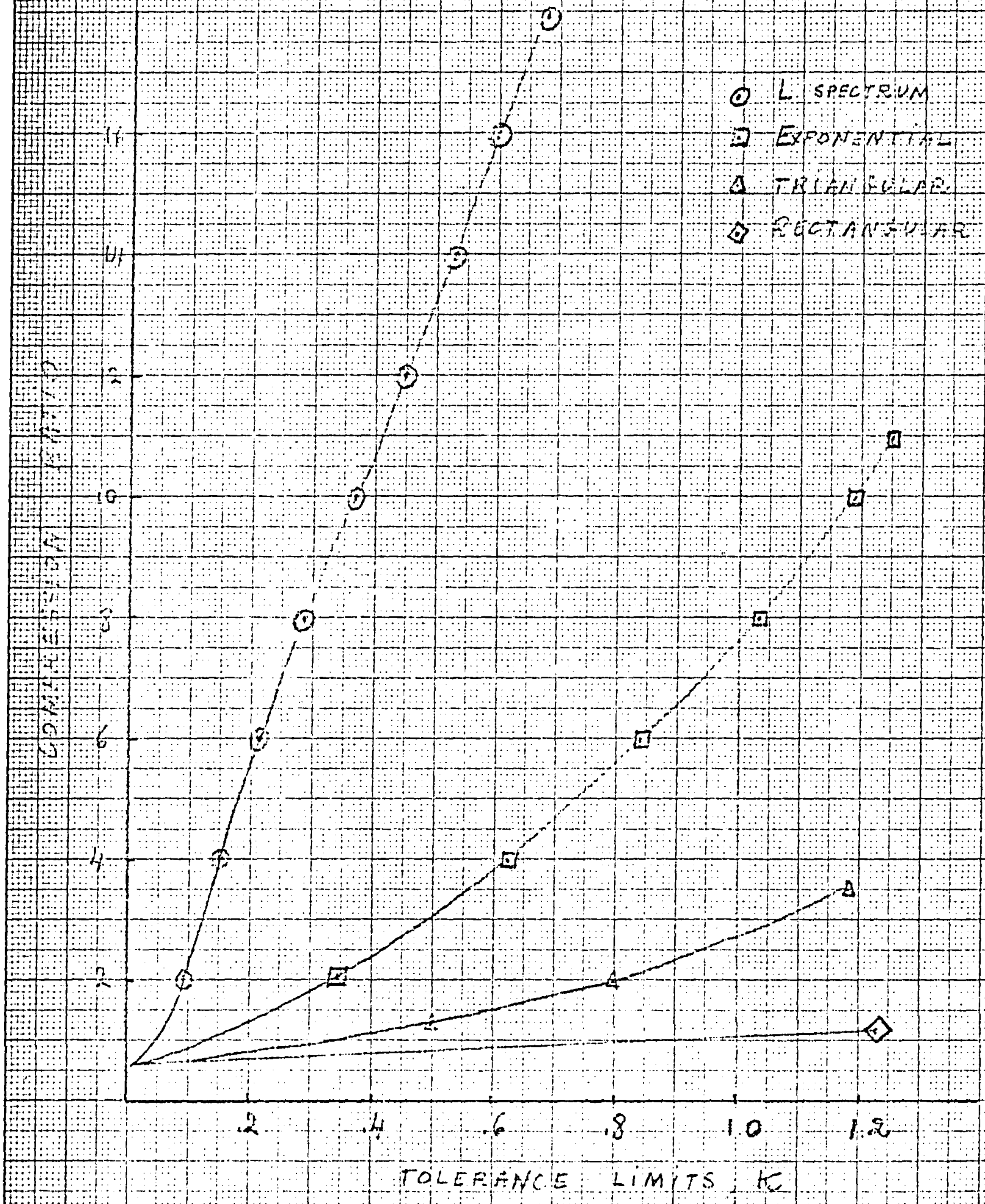


FIG 4.3 COMPRESSION RATIO FOR A FIRST-ORDER PREDICTOR

(FROM M.M. BRUCE, REF. 46 p 9)

Davisson has shown that its performance improves for data having statistical dependence beyond the previous sample, as is the case for a second-order Markov process. These data can be approximated by a sequence of non-level straight line runs and one expects that first-order schemes would have some advantage.

Denote by p_γ the probability that a jump in the amplitude of two adjacent samples is followed by a jump of the same magnitude, within a given tolerance γ.

$$p_\gamma = \Pr(\text{Jump of size } j \text{ followed by jump of size } j \pm \gamma)$$

Then the probability of a run of length n is given by (we drop the subscript)

$$(4.3) \qquad p(n) = p^n(1-p)$$

and the average length of the runs is

$$E(n) = \sum_n np^n(1-p)$$

Recognizing that

$$\sum_n np^n = p\frac{d}{dp}\left(\sum_n p^n\right)$$

we obtain

$$(4.4) \qquad E(n) = (1-p)\sum_n np^n = \frac{p}{1-p}$$

Therefore

$$C_s = E(n) = \frac{p}{1-p}$$

If C_s is the only parameter that can be measured, we can then calculate the value of p

$$(4.5) \qquad p = \frac{C_s}{C_s + 1}$$

Note that this value of p is approximate and implies the following property of the process; p does not depend on the size of the initial jump. Although this may fit certain telemetry data, it does not describe TV signals. In the following, we assume that (4.3) holds for the run length distribution.

4.1.2 Aperture error

We can proceed now to calculate the aperture error given by (4.1). We have

$$E(\gamma/n) = \frac{m(m+1)}{3 \cdot 2^{2k}} \frac{n-1}{n} \qquad \text{(see (3.6))}$$

and since

$$\sum_{k=0}^{n} k^2 = \frac{n(n+1)(2n+1)}{6}$$

we can write

$$E(i^2/n) = \frac{1}{n} \sum_{k=0}^{n-1} i^2 = \frac{(n-1)(2n-1)}{6}$$

Substituting the value of $E(i^2)$ into (4.1), we obtain

$$(4.6) \qquad E(e_A^2/n) = \frac{m(m+1)}{3\ 2^{2k}} \frac{n-1}{n} \frac{6n^2+(n-1)(2n-1)}{6(n-1)^2}$$

Therefore

$$E(e_A^2) = \frac{m(m+1)}{3\ 2^{2k}} \sum_{n=1}^{2^r-1} \frac{(n-1)6n^2+(n-1)(2n-1)}{6n(n-1)^2} p(n)$$

or

$$(4.7) \qquad E(e_A^2) = \frac{m(m+1)}{3\ 2^{2k}} \sum_{n=1}^{2^r-1} \frac{6n^2+2n-1}{6n(n-1)} p^n(1-p)$$

When m = 0, we have $E(e_A^2) = 0$; thus there is no error between the reconstructed and original quantized value.

A computer calculation of (4.7) has shown that this expression does not depend strongly on C_s when $2 < C_s < 10$.

4.2 Effect of Transmission Errors

As for the zero-order predictor, channel noise results in amplitude error and timing error.

4.2.1 Level word error

We assume that a single error in a level word is equally likely in each bit. An error in level word j affects the reconstructed samples in run j and run j+1, as shown in Figure 4.3

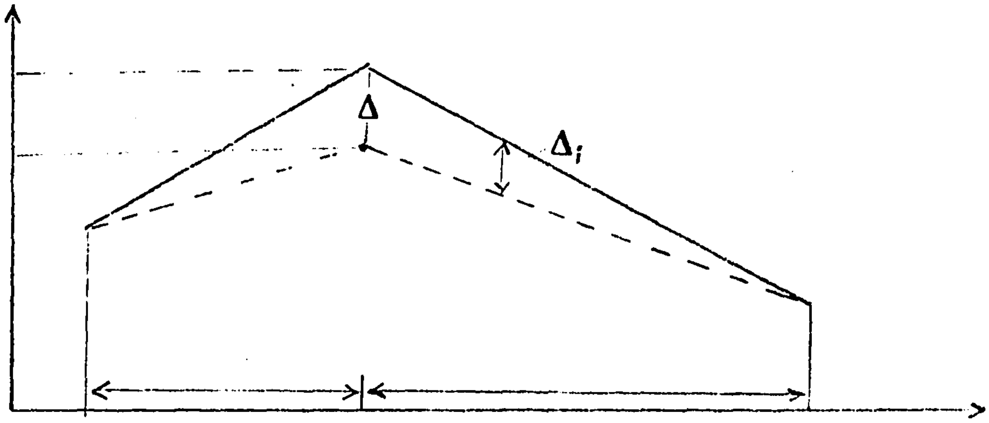


Figure 4.4 First-Order Level Word Error

Let X_i denote the amplitude of the i^{th} nonredundant sample S_i and X_i' the amplitude of the sample in error. In Figure 4.4, sample S_j is erroneous since its amplitude is X_j' instead of X_j. This error is propagated over (n+n'-1) samples and on the average the error is propagated over $2C_s$-1 samples.

Now, given an error of Δ steps ($0 \leqslant \Delta \leqq 2^k$) in sample S_j, an error Δ_i will result for all redundant samples from S_{j-1} to S_j. Moreover,

$$\Delta_i < \Delta \qquad \text{for } i = 1, \ldots\ldots, n-1$$

since the nonredundant sample S_j is the n^{th} sample in the run and clearly displays the largest error, as shown in Figure 4.4.

In fact, since the errors $_i$ decrease linearly when i varies from (n-1) to 1 (at i=1 we have $\Delta_i=0$), we can write

$$E(\Delta_i) = \Delta/2.$$

If these were the only errors resulting from an erroneous nonredundant sample, the m.s. error resulting from a first-order scheme would be half that of a zero-order predictor, hence, half that of a PCM (non-compressed) system. However, as illustrated in Figure 4.2, the error propagates to the next adjacent run. For this run the errors Δ_i' $(i=1, 2, \ldots\ldots, n'-1)$ are also smaller than Δ and independent of the run length n', and their average value is

$$E(\Delta_i') = \Delta/2$$

Consequently, given an error Δ in sample S_j, the average error for the resulting (n+n'-1) erroneous samples of a first-order predictor is equal to Δ.

The m.s. error per sample is, therefore, the same as that of a zero-order system (or that of a non-compressed system) and is given by

$$(4.9) \qquad E(e^2/\text{error in level word}) = E(X_j - X_j'/\text{error}) = \frac{1}{3k}\cdot\frac{2^{2k}-1}{2^{2k}}$$

Since Pr(error in level word) $\cong kP_B$, we have

$$(4.10) \qquad E(e_A^2) = \frac{P_B}{3}\cdot\frac{2^{2k}-1}{2^{2k}}$$

4.2.2 Timing word error

The m.s. error of the reconstructed waveform depends on the addressing scheme used to inform the receiver on the location of the nonredundant samples. We shall consider here the case of run length encoding. Every level word is accompanied by a timing word which indicates the distance d separating two adjacent nonredundant samples. The receiver reconstructs the data by joining with a straight line the two samples d units apart. Note that for all non-level runs, d can be constituted of at most 2^k steps, where k is the length of an amplitude word, since this would correspond to a signal starting from 0 volt and reaching its maximum value of 1 volt (we recall that the signal has been normalized).

Figure 4.5 illustrates the effect of an error in a run-length word.

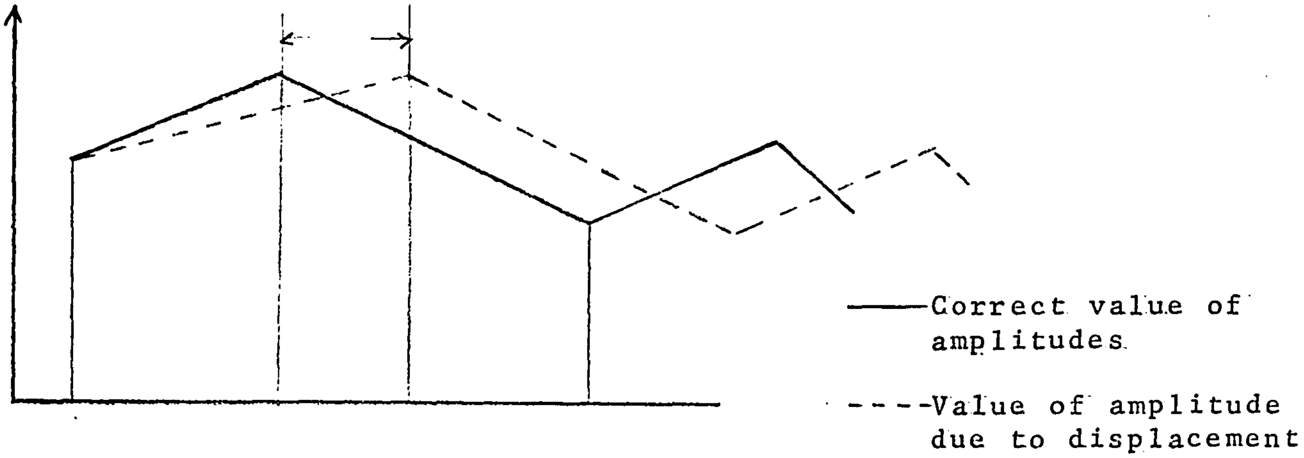

——Correct value of amplitudes
----Value of amplitude due to displacement

Figure 4.5 Run-Length Word Error

Due to an error in the (n-1)th run-length word, sample S_n is displaced and becomes now S'_n. In this example, the distance between S'_n and S_n is taken as d=5. All succeeding nonredundant samples are displaced by the same amount. If we denote by n_r the number of affected nonredundant samples, we have, similarly to the zero-order case

$$(4.11) \qquad \frac{E(n_r)}{L/C_s} = \frac{1 + C_s/L}{2}$$

The error due to sample displacement varies from 0 step to 2^k-2^x steps, where x is such that $2^x/2^k$ is the amplitude of sample S_n. Here x is a r.v. assumed to be uniformly distributed between 0 and (k-1). Thus,

$$(4.12) \qquad \Pr(x = i) = 1/k \; ; \quad i = 0, 1, \ldots, k-1$$

Assuming also that the error has the conditional uniform probability

$$P(e/x) = \frac{1}{2^k-2^x} \;,$$

it is possible to write the following expression for the conditional m.s. error (note that the error can assume the following values $e = j/2^k$ where $j = 0, 1, \ldots, 2^k-2^x$).

$$E(e^2/x,\text{error}) = \frac{1}{2^k(2^k-2^x)} \sum_{j=0}^{2^k-2^x} j^2$$

This expression simplifies to

$$E(e^2/x,\text{error}) = \frac{1}{6\;2^k(2^k-2^x)} (2^k-2^x)(2^k-2^x+1)(2^{k+1}-2^{x+1}+1)$$

or

$$E(e^2/x,\text{error}) = \frac{(2^k-2^x+1)(2^{k+1}-2^{x+1}+1)}{6\;2^k}$$

Since

$$E(e^2/\text{error}) = \sum_x E(e^2/x,\text{error}) \Pr(x) \;,$$

we obtain from (4.12)

$$(4.13) \qquad E(e^2/\text{error}) = \frac{1}{6k \cdot 2^k} \sum_{x=0}^{k-1} (2^k - 2^x + 1)(2^{k+1} - 2^{x+1} + 1)$$

Denoting the expression under the summation by f(k,x), we have

$$(4.14) \qquad E(e^2/\text{error}) = \frac{1}{6k \cdot 2^k} \sum_{x=0}^{k-1} f(k,x)$$

Now it is evident from Figure 4.1 that all redundant samples except one (sample S_i) are affected. Hence, if n_s is the number of affected samples per run, we have

$$\frac{E(n_s/\text{error})}{C_s} \cong \frac{C_s - 1}{C_s} .$$

Since

$$E(e^2) = \frac{E(n_r/\text{error})}{L/C_s} \cdot \frac{E(n_s/\text{error})}{C_s} E(e^2/\text{error}) \Pr\binom{\text{one timing}}{\text{error}}$$

and

$$\Pr\binom{\text{one timing}}{\text{error}} \cong rP_B ,$$

we may write

$$(4.15) \qquad E(e^2) = \frac{1 + C_s/L}{2} \cdot \frac{rP_B(C_s - 1)}{C_s} \cdot \frac{1}{6k \cdot 2^k} \sum_{x=0}^{k-1} f(k,x)$$

The overall r.m.s. error in the reconstructed data is therefore (assuming no aperture error, i.e., m = 0)

$$(4.16) \qquad e_{rms} = \left\{ \frac{1}{12\ 2^{2k}} + \frac{P_B}{3} \frac{2^{2k} - 1}{2^{2k}} + \frac{1 + C_s/L}{2} \cdot \frac{rP_B(C_s - 1)}{C_s} \cdot \frac{1}{6k\ 2^k} \sum_{x=0}^{k-1} f(k,x) \right\}^{\frac{1}{2}}$$

where $f(k,x) = (2^{k}-2^{x}+1)(2^{k+1}-2^{x+1}+1)$

This equation is plotted in Figure 4.6 for $k = 6$ bits, $r = 5$ bits. It is seen that the r.m.s. error increases exponentially when $P_B > 10^{-6}$, and does not vary appreciably with C_s

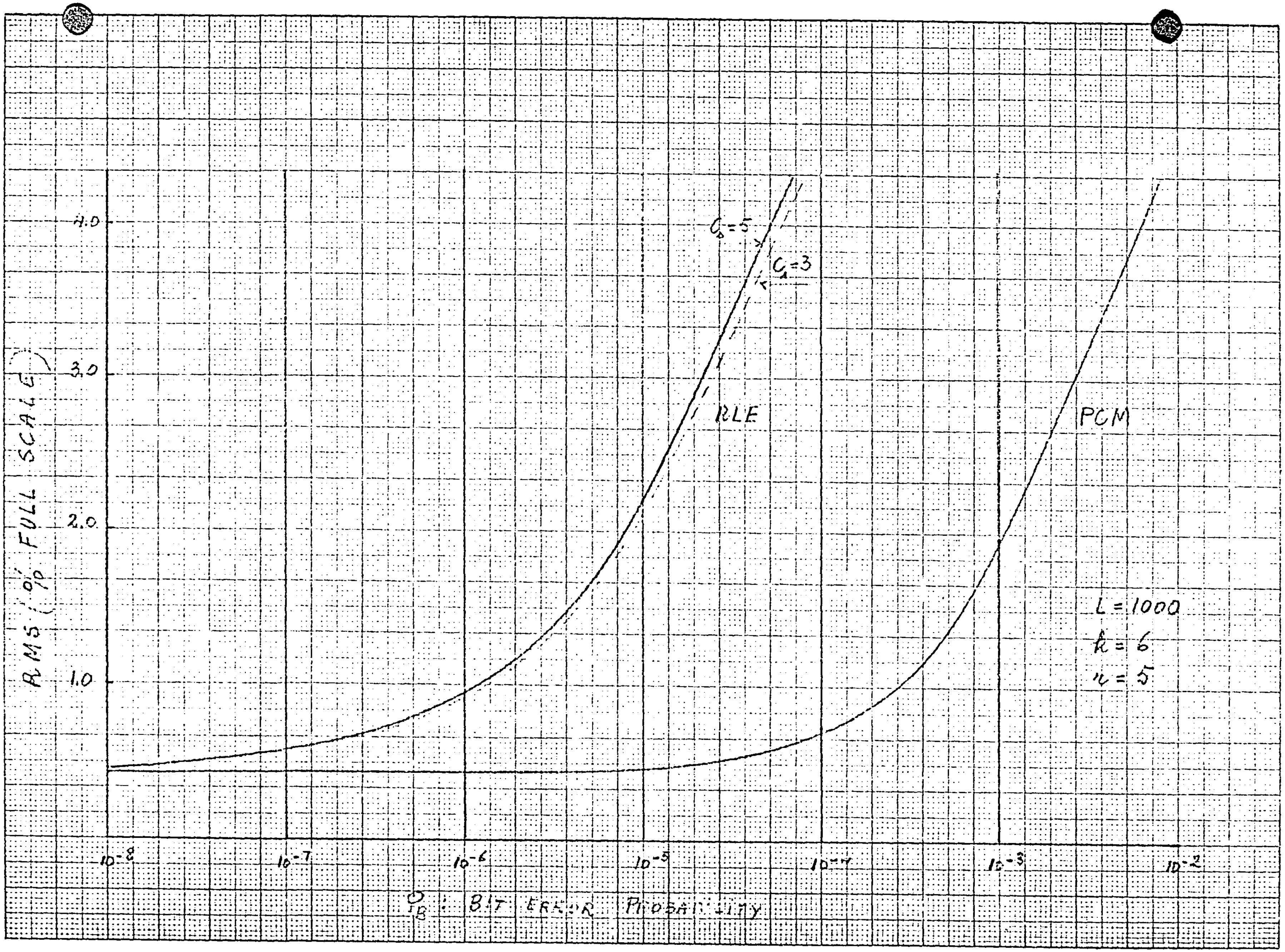

RMS (% FULL SCALE)
4.0
3.0
2.0
1.0
$C_s=5$
$C_s=3$
RLE
PCM
L = 1000
k = 6
n = 5
10^{-8}
10^{-7}
10^{-6}
10^{-5}
10^{-4}
10^{-3}
10^{-2}
P_B : BIT ERROR PROBABILITY

Figure 4.6

CHAPTER V

CONCLUSION

In the course of this study, we have seen how the redundancy reduction can be obtained by various forms of coding. A statistical analysis of the effects of channel noise on zero-order and first-order predictors has been presented in Chapters III and IV.

We showed that, in the absence of correlation, the vulnerability of the signal increases. It is apparent from the curves illustrated by figures 3.7 and 4.6 that, given a channel with a bit error probability of 10^{-7}bits/sec or more, the resulting r.m.s. error of the compressed systems may not be acceptable. However, of the three addressing techniques considered, the position word encoding technique yields the smallest error in the reconstructed data. Moreover, with this scheme, the error propagation is limited to one run.

We have seen that more elaborate coding methods can be devised; however, it appears that efficient encoding often implies excessive complexity of the transmitting and receiving equipments, and much work remains to be done to find a "practical" optimum coding.

It should be emphazised here that the data compression systems studied in Chapters III and IV are implemented in such a way as to yield both bandwidth compression and energy compression. Thus, the average compression ratio C_s defined in this study implies that the compressed data require a bandwidth B/C_s and signal energy E/C_s, where B and E denote the bandwidth and signal energy for the non-compressed system, respectively.

For certain applications, such as telemetry in deep space, the energy compression is a prime requirement. On the other hand, in many applications, such as videophone, it may be more important to reduce the bandwidth rather than the power. Therefore, if the power E used for the non-compressed system is available for the compressed system, the comparison between the two systems must be made under the assumption of equal energy.

The bit-error probability of the non-compressed system is given by

$$P_B = \exp(- \frac{2S}{N_0 R})$$

where R is the transmission bit rate.

The bit rate of the compression system is R/C_s. Thus the bit-error probability P_B' of the compression system is

$$P_B' = \sqrt{\frac{N_0 R \pi}{2S}} \exp(- \frac{2SC_s}{N_0 R})$$

Replacing this value of P_B in (3.36), (3.8) and (4.16)

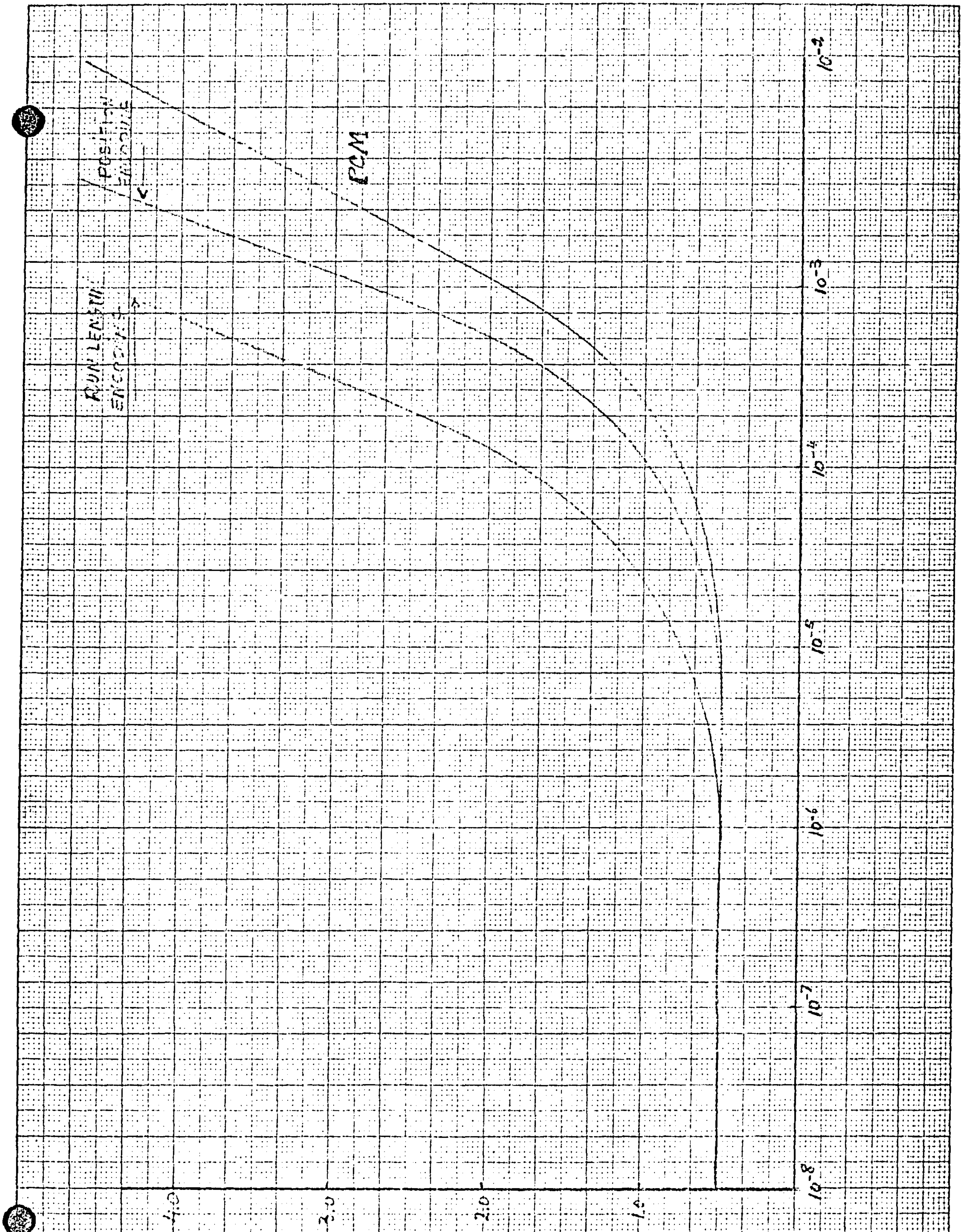

RUN LENGTH ENCODING
POSITION ENCODING
PCM
4.0
3.0
2.0
1.4
10^{-8}
10^{-7}
10^{-6}
10^{-5}
10^{-4}
10^{-3}
10^{-2}

Figure 5.1

yields the new equations for the r.m.s. error of the compression system. The results are plotted in Figure 5.1 and show clearly that for practical channels, bandwidth compression can be achieved with only a small signal degradation compared to PCM transmission.

When the power necessary for PCM transmission is not available, one can resort to error-correcting codes to reduce the error rate. Several coding procedures have been described in the literature (48) (53). It is shown that protection on the address words only can be sufficient.

Block coding minimizes the bit error rate but not necessarily the magnitude of the m.s. error, and therefore is not efficient for most compression systems. Majority vote coding (51) yields better results. It is possible that "significant bit" coding could provide an efficient way to cope with this problem. This coding procedure consists in protecting the most significant bits of a word thus minimizing the errors which are the most costly.

However, any form of coding results in lowering the bit compression ratio and an optimum procedure has yet to be found.

BIBLIOGRAPHY

CHAPTER I

1. D.R. Weber, "A Synopsis on Data Compression", Proc. NTC; 1965.

2. L.D. Davisson, "Information Systems", Goddard Summer Workshop report; 1966.

3. B.M. Oliver, "Efficient Coding", Bell Systems Tech. J., Vol. 31, July 1952.

4. T.J. Goblick, "Theoretical Limitations on the Transmission of Data from Analog Sources", IEEE Trans. on Information Theory, Vol. IT-11, October 1965.

5. H. Blasbalg and R. Van Blerkom, "Message Compression", IRE Trans. on Space Electronics and Telemetry, September 1962.

6. D.H. Shaefer, "Logarithmic Compression of Binary Numbers", Proc. IEEE, Vol. 49, July 1961.

7. R.M. Karp, "Minimum-Redundancy Coding for the Discrcte Noiseless Channel", IEEE Trans. on Information Theory, Vol. IT-7, January 1966.

8. Proc. of the IEEE; special issue entitled "Redundancy Reduction", March 1967.

9. L.W. Gardenhire, "Redundancy Reduction the Key to Adaptive Technique", Proc. NTC; June 1964.

10. G.A. Rahe, "A Comparison of Several Types of Data Compression", Ph.D. Dissertation, UCLA, 1965.

11. S.A. Shelkahl, "Comparison of Hardware Requirements for Polynomial Data Compressors"; Proc. NTC, 1968.

12. L.D. Davisson, "The Theoretical Analysis of Data Compression Systems"; Proc. IEEE, February 1968.

13. L.D. Davisson, "Data Compression Using Straight Line Interpolation", IEEE Trans. on Information Theory, IT-14, May 1968.

14. R.L. Kutz, J.A. Sciulli and R.A. Stampfl, "Adaptive Data Compression for Video Signals", in "Advances in Communication Systems", Academic Press, Vol. 3, 1968.

15. C.W. Harrison, "Experiments with Linear Prediction in Television", Bell Systems Tech. J., July 1952.

16. J.W. Schwartz, R.C. Barker, "Bit-Plane Encoding", IEEE Trans. on Aerospace and Electronics Systems, July 1966.

17. G.R. Schwarz, "Buffer Design for Data Compression Systems", IEEE Trans. on Communication Technology, Vol. COM-16, August 1968.

18. J.E. Medlin, "The Prevention of Transmission Buffer Overflow in Telemetry Data Compressors", IEEE Trans. on Communication Technology, Vol. COM-16, February 1968.

19. H. Massey, "An Experimental Telemetry Data Compressor", Proc. NTC, 1965.

20. A.V. Balakrishnan, "An Adaptive Nonlinear Data Predictor", Proc. NTC, 1962.

CHAPTER II

21. D. Estournet, "Compression d'information de signaux d'images par les systemes differentiels codes"; l'Onde Electrique, September 1969.

22. A.J. Sapler, "The Coding of Visual Signals to Reduce Channel-Capacity Requirements"; Proc. IEE, July 1962.

23. L.E. Franks, "A Model for the Random Video Process"; Bell Systems Tech. J., April 1966.

24. D. Estournet, "Etude statistique d'un signal d'image", l'Onde Electrique; September 1969.

25. E.R. Kretzmer, "Statistics of Television Signals"; Bell Systems Tech. J., July 1952.

26. A. Karpel, "Measurements in the Power Spectrum of TV Signals", (Australia PMG's Dept. Report No. 4632) September 1957.

27. Deriugin, "The Power Spectrum and the Autocorrelation of the Television Signal", Telecommunications, Vol. 7, 1957.

28. P. Elias, "Entropy and Autocorrelation", Proc. of the IRE, July 1951.

29. S. Narayanan and L.E. Franks, "The Spectra of Digitally Encoded Video Signals", Int'l. Conf. on Communication, Denver, June 1969.

30. T.S. Huang, "PCM Picture Transmission", IEEE Spectrum, December 1965.

31. C. Cherry et al., "An Experimental Study of the Possible Bandwidth Compression of Visual Image Signals", Proc. IEEE, Vol. 51, November 1963.

32. R.E. Graham, "Subjective Experiments in Visual Communication", IRE Nat'l. Conv. Rec., Vol. 6, Pt. 4, pp. 100-106, 1958.

33. A.J. Seyler, "Visual Communication and Psychophysics of Vision", Proc. of IRE (Australia), 1962, 23, p. 291.

34. L.G. Roberts, "Picture Coding Using Pseudo-Random Noise", IRE Trans. Vol. IT-8, February 1962.

35. W.F. Schrieber, "Picture Coding", IEEE Proc. March 1967.

36. D.P. Peterson and D. Middeton, "Sampling and Reconstruction of Wave-Number-Limited Functions in n-dimensional Euclidian Spaces", Information and Control, Vol. 5, 1962 (pp. 279-323).

37. T.S. Huang and M. Chikhaaui, "The Effect of BSC on PCM Picture Quality", IEEE Trans. Vol. IT-13, April 1967.

38. D.N. Graham, "Optimum Filtering to Reduce Quantization Noise", M.S. thesis, MIT, January 1962.

39. A.J. Seyler and Z.L. Budrikis, "Detail Perception after Scene Changes in Television Image Presentations", IEEE Trans. on Information Theory, Vol. IT-II, January 1965.

40. A.J. Seyler, "Statistics of Television Frame Differences", IEEE Proc. December 1965.

CHAPTER III

41. S. Karp, "Noise in Digital-to-Analog Conversion Due to Bit Errors", IEEE Trans. Vol. SET-10, No. 3, September 1964.

42. A.J. Viterbi, "Principle of Coherent Communication" McGraw-Hill Co., 1966.

43. T.L. Belver, "Effect of Transmission Errors in Image Data Compression Techniques", Proc. NTC Record, 1969.

44. L.D. Davisson, "The Concept of Energy Compression Ratio and Its Applications to Run Length Coding", Final Report, 1966 Session of the Goddard Summer Workshop.

45. R.B. Kelley, "The Effect of Channel Errors on a Sample Compression System Employing Binary Sequence Time Encoding", Final Report, 1966 Session of the Goddard Summer Workshop.

CHAPTER IV

46. M.M. Bruce, "An Analysis of a First-Order Predictor for Data Compression", NASA TN D-3866.

47. L.D. Davisson, "The Fan Method of Data Compression", Final Report, 1966 Session of the Goddard Summer Workshop.

48. G.L. Raga, "Channel Noise - A Limiting Factor on the Performance of a Class of Adaptive Techniques", Conference on Adaptive Telemetry, Goddard Space Flight Center, 1967.

49. T.J. Lynch, "Data Compression with Error-Control Coding for Space Telemetry", NASA TR R-261.

50. L.D. Davisson, "The Effect of Channel Errors on Data Compression", WESCON Conf. (Session 6), 1967.

51. M. Kanefsky, "Data Compression Using Bit Plane Encoding", Conference on Adaptive Telemetry, Goddard Space Flight Center, 1967.

52. F.W. Mounts, "A Video Encoding System Using Conditional Picture-Element Replenishment", Bell Systems Tech. J., Vol. 48, September 1969.

53. J.O. Limb and F.W. Mounts, "Digital Differential Quantizer for Television", Bell Systems Tech. J., Vol. 48, September 1969.