X-521-65-320

N66-11218

(ACCESSION NUMBER) (THRU)

24 1

(PAGES) (CODE)

07

(NASA CR OR TMX OR AD NUMBER) (CATEGORY)

FACILITY FORM 602

NASA TM X-55323

A DATA COMPRESSION PRIMER

BY

GERALD HOGG

GPO PRICE $

CFSTI PRICE(S) $

Hard copy (HC) 1.00

Microfiche (MF) .50

ff 653 July 65

AUGUST 1965

NASA

GODDARD SPACE FLIGHT CENTER

GREENBELT, MARYLAND

X-521-65-320

A DATA COMPRESSION PRIMER

by

Gerald Hogg

August 1965

Goddard Space Flight Center
Greenbelt, Maryland

CONTENTS

LIST OF ILLUSTRATIONS

A DATA COMPRESSION PRIMER

INTRODUCTION

There have been many papers written about data compression but there appears to be a lack of short papers designed to familiarize the uniniated with this rapidly growing field. Unfortunately, some of these papers differ in the terminology used, making it difficult for the newcomer to quickly develop a background of data compression nomenclature. Without a background of the terminology used, the reader may become confused about the techniques used. This report was written for the purpose of explaining the operation of data compression systems to those who would like to gain knowledge of the methods which are under study. No new techniques or ideas relating to data compression will be discussed in this report. Instead, an attempt will be made to explain the general methods used and to establish a standard data compression terminology. Appendix B contains a glossary of data compression terms which may be helpful in reading this report.

DATA GATHERING SYSTEM

The typical spacecraft telemetry system in use today incorporates a commutator to sample individual data sources. Each source or channel is sampled at fixed intervals equal to the product of the time necessary to sample each individual channel and the number of channels to be sampled. The sampling rate is fixed, although in many cases the information rate varies from low activity to high activity. At low data activity the now relatively high sampling rate produces much redundant data.

A distinction must be made between "data" and "information." Data is the carrier of information and the medium at which an observer looks to acquire information. If less data can be sent without decreasing the amount of information it contains, then the burden upon the telemetry system can be reduced, possibly enough to increase efficiency, depending on trade-offs in power, size, weight and range, without sacrificing the experiment mission. The goal of any data compression system is to eliminate as much redundancy as possible in order to "compress" bandwidth.

ENTROPY REDUCING TRANSFORMATIONS

Each experiment or status monitor sensor may be sampled directly by the commutator, and in fact, many are. However, an experimenter may wish to do some signal conditioning or preprocessing and so design his experiment to accomplish this end. Many times this signal conditioning results in a type of data compression called Entropy Reduction.[1] Table 1 contains a list of some of the more common entropy reduction data compression devices. The data gathering system now in general use for satellite experiments is shown in Figure 1.

Table 1
SOME ENTROPY REDUCTION DATA COMPRESSION METHODS COMMONLY USED ON SCIENTIFIC SATELLITES

1. Logarithmic Amplifiers	7. Second Moment Estimators
2. Digital Counters	8. Variance Estimators
3. Floating Point Counter	9. Quantizers
4. Limiters/Clippers	10. Filters (Low Pass, High Pass, Band Pass)
5. Threshold Monitors	11. Probability Distribution Fit Estimators
6. Mean Estimators	

Entropy reduction (ER) devices enable the transmission of more information per bit than would be possible with normal instrumentation. These ER devices act as a <u>non-reversible transformation</u> on the input data. Using a non-reversible transform results in a pre-transmission loss of some of the original data, which cannot be recovered after transmission.

The nature of the sampled data system is such that if a data source has not changed from a particular value for any length of time, that value will be sent repetitively. The number of times a value is sent depends upon the sampling rate of the commutator and the period of source inactivity. Quite often the electronics of a sensor require a recycling time which is large with respect to the sampling rate of the telemetry commutator. The result of this is the transmission of data values to the ground which do not differ significantly enough from one another to constitute a transfer of information comparable to the amount of data transmitted. Although ER devices help to reduce this redundancy, a great number of redundant samples are still transmitted. Of course, some experiments cannot be instrumented using an ER device due to the nature of the measurements being attempted. Then too, new experiments need operational data to insure proper design improvement for future flights.

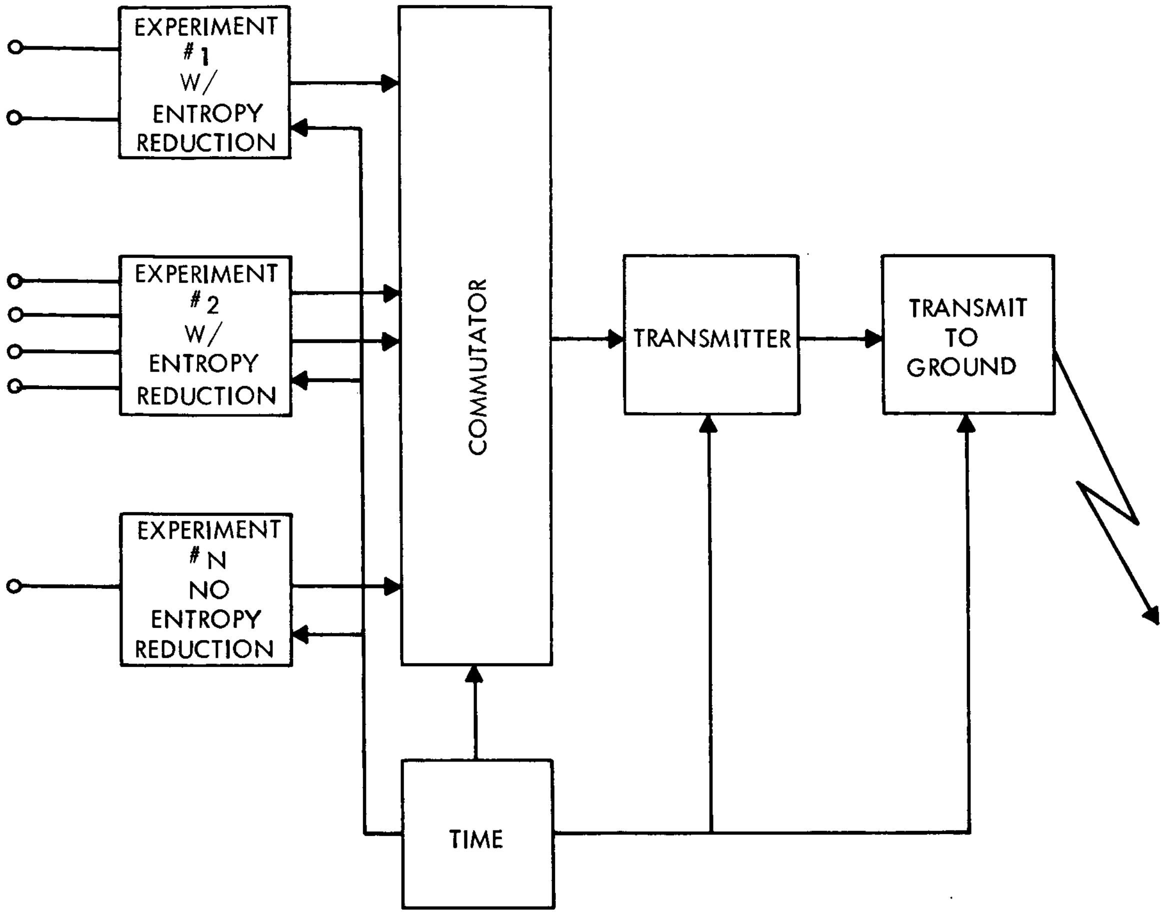


Figure 1—Typical On-Board Satellite Telemetry System

INFORMATION PRESERVING TRANSFORMATION

A way to accomplish compression on repetitive data or sources where the waveform is important is with a method of redundancy removal which employs a reversible transformation. The type of data compression devices which fit in this class are known as Information Preserving (IP) devices. Although the data compression devices in this class are in the ER group in a strictly classical sense[2], because they do reduce entropy (H) and some of the original data is lost, the process is reversible to within a specified allowable tolerance or peak error.

There are many types of IP data compression devices. One class of IP data compression which has been reported in the literature to a great extent is that of polynomial data compression techniques.[3]

The polynomial data compressors approximate the original data to within a specified peak error. This maximum allowable error is called the tolerance. The polynomial data compression schemes considered most promising are listed below:

1. Zero-Order Predictor
2. Zero-Order Interpolator
3. First-Order Predictor
4. First-Order Interpolator

These four algorithm have been investigated[4] by means of computer simulation and can be rated from their performance on a generalized waveform. The algorithms named above are increasingly better (for a generalized waveform) as you go down the list. The first-order predictor is unsatisfactory for noisy data however, since it will tend to oscillate when small fluctuations occur in the data.

Second-order polynomial algorithms have also been examined by means of computer simulation and have also proven to be unsatisfactory for general use as they too exhibit tendencies to oscillate when operating on noisy data. These second-order algorithms also have proven to be undesirable because of inherent design complexity.

As a general rule, the type of algorithm to use in a particular application is highly dependent on the data characteristics of the source. A zero-order predictor, because it is simple, is better for compressing a step function, such as a square wave, than a first-order algorithm. If a signal of some other type, such as a sawtooth, were in the same telemetry system, however, a zero-order predictor might not be as desirable as a first-order algorithm but could still be used. Because of the many trade-offs involved, close examination of the total system, including data sources, must be made before a data compression system can be implemented.

A description of each compression algorithm listed above is included in Appendix A along with accompanying illustration for each type of algorithm.

COMPRESSION RATIO

A common parameter in use for measuring compression efficiency is the word compression ratio. This parameter is defined to be:

$$C_r = \frac{W_i}{W_o}$$

W_i = number of data points input to the IP data compression algorithm,
W_o = number of data points output from the IP data compression algorithm.
The word compression ratio does not take into account any additional bits which must be added to the data point prior to transmission. A system or bit compression ratio is used to describe total system efficiency.

A measure of the amount of relative redundancy (R) in a source is defined to be:

$$R = 1 - \frac{1}{C_r}$$

where C_r is the word compression ratio.[5]

When comparing data compression systems using a compression ratio criteria, care must be taken to be sure that the sampling rates of the sources being compared are equivalent. If not, the system with the highest equivalent sampling rate will show the best compression ratio as it rejects the additional redundant points due to the extra over sampling.

BUFFER MEMORY

When an IP data compression system is added to the telemetry system, the relationship between the data and its channel and the time information changes from that in the present day telemetry system. The data is input to the data compression algorithm from the computer in a synchronous manner. After the reversible transfer takes place, some of the channels of a frame will not be output. The result is an asynchronous output from the data compression algorithm. By asynchronous we mean the output appears in a seemingly random fashion and the synchronous telemetry frame is eliminated.

A major subsystem of an IP data compression system is the Buffer. The buffer is needed to reestablish a synchronous bit rate so that standard telemetry systems can be used for transmission of compressed data. The buffer, when implemented into a hardware system, must be some sort of storage device such as a magnetic core memory. The memory will have a finite storage capability. If the average read-in rate/read-out rate goes close to one, such as during an active data burst, then many channels will have outputs from the compression algorithm, and the probability of buffer overflow will be large. Buffer overflow is the term used to describe the condition of full buffer memory. Any additional inputs to the memory cannot be stored and are lost – hence overflow. The loss of any data, especially at a time of some important event, cannot be allowed. When it is recalled that one compressed data point represents many uncompressed data points, the solution of the problem of buffer overflow clearly becomes mandatory.

There are several ways of controlling buffer overflow. One method commonly reported in the literature is adaptive aperture control.[6] This type of buffer control operates by sensing either buffer fullness or data activity or both. Depending upon the fullness or amount of data activity, the tolerances within the data compresion algorithm are increased in order to decrease the number of significant samples output from the data compression algorithm to the buffer. An obvious disadvantage of this method is the increase of peak error at an active period when the data may be of the most interest to the experimenter. The philosophy used to justify this method implies that, because of the higher activity, the peak errors are not as noticeable due to steep slopes and sudden changes when viewed by the human eye assuming human data examination.

A second method of buffer control is called adaptive sampling.[7] With this type of adaptive control the buffer changes the sampling sequence in the commutator to reduce the sampling rate of active channels if overflow is eminent. If a randomly addressable memory was used for a commutator, this could be a completely flexible operation as any sampling sequence could be set up.

Another method of buffer control is that of adaptive filtering.[8] Here the data is sampled near the frequency of the system noise. An average is taken of N points, thus smoothing the data. This average is transmitted to the data compression algorithm and from there on to the buffer. If data activity increases, tending to cause the buffer fullness condition, the number of samples averaged, N, will be increased, thus decreasing the total number of points input to the data compression algorithm and buffer.

Both adaptive filtering and adaptive sampling methods of buffer control are in their development stage. Very little information on this subject has appeared in the literature, however, a great deal of study effort is being expended on these techniques.

IDENTIFICATION METHODS

Because of the elimination of synchronous frames, each data point stands alone and some means of identifying each telemetry channel is needed. The common way of doing this is to add enough bits to the data word to enable a code large enough to identify each channel.

TIMING METHODS

A need for some means of time code also arises because of the elimination of the synchronous telemetry frame format. This time code is necessary due to the time delays in a IP data compression system which would cause loss of time information. These delays are:

1. Fixed delay between sample time and output from the data compression algorithm.
2. Delay in the data compression buffer.
3. Delay in the reconstruction device, if used, at the receiving station.

Several methods of determining the time at which a sample was taken have been developed. One system which might be used, if a system clock is available, operates by first sending a time word at the beginning of each major frame. A short code is then included at the beginning of each minor frame. The time of any channel occurring between these latter codes can be extrapolated by using knowledge of the sampling rate and channel ID code number. Figure 2 represents the format which would arise from usage of this time coding method.

BIT PLANE ENCODING

Bit plane encoding[9] is a method of IP or ER data compression, depending on the coding method used, based upon the supposition that lower ordered bits of data samples change more frequently than higher ordered bits of data samples. The name "Bit Plane" is derived from the manner in which data value are stored in a magnetic core memory. The values are stored in columns of a memory cube with bits of equal magnitude contained in "bit planes." Bit plane encoding is accomplished by reading words into the columns of the memory, then reading out and encoding the planes. The encoding schemes used depend upon the magnitudes and activities represented by the particular bit planes. Very high order planes are not transmitted unless significant changes occur. Similarly, very low order planes are not transmitted because variations are normally due to noise and contain no significant information.

MINOR FRAME NO.									
1	FRAME SYNC	TIME CODE		CHAN 2	3	7	11	13	14
2	FRAME SYNC	TIME MARK	CHAN 1	5	8	15	2	10	1
3	FRAME SYNC		CHAN 4	6	12	14	16	7	11
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
N	FRAME SYNC	"	-	-	-	-	-	-	-
1	FRAME SYNC	TIME CODE		CHAN n_i	-	-	-	-	-
2	"	TIME MARK	-	-	-	-	-	-	-
3	"	"	-	-	-	-	-	-	-
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
N	"	"	-	-	-	-	-	-	-

Figure 2–Frame Format W/IP Data Compression Time Codes

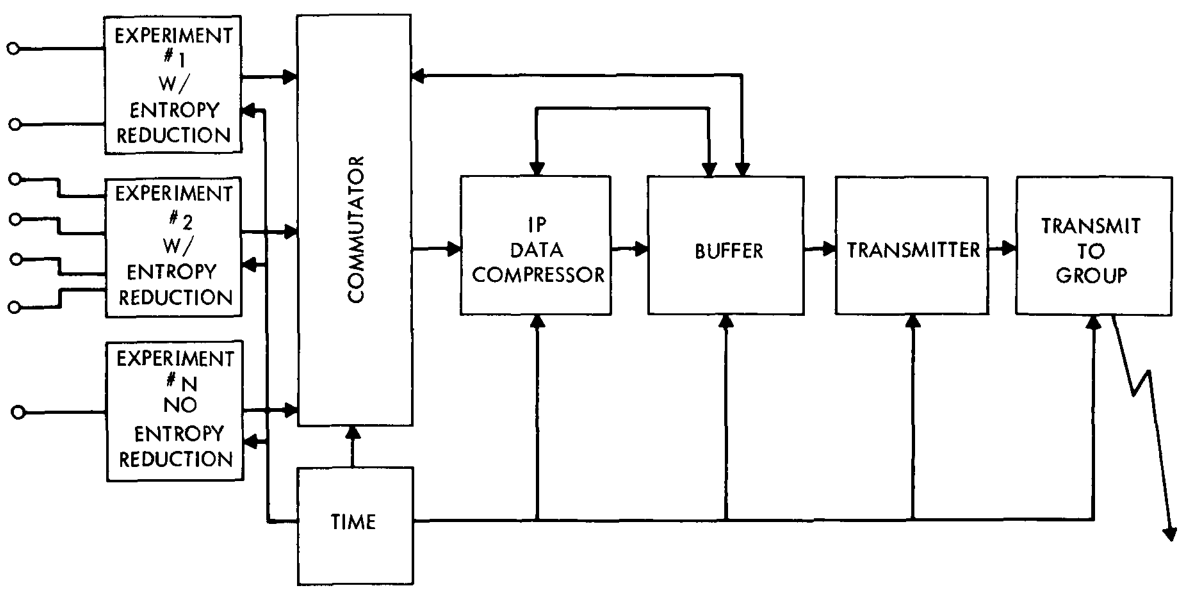


Figure 3–On-Board Satellite Telemetry System W/IP Data Compression

The problems of channel identification, time correlation, and buffer overflow found in polynomial compression systems are also inherent in bit plane encoding systems. Little has been reported on the differences and peculiarities of these problems in relation to bit plane encoding.

Figure 3 illustrates the spacecraft data handling system with both ER and IP data compression included.

ERROR CONSIDERATIONS

Transmission link errors are overcome at the present time by virtue of the redundancy inherent in the data. With data compression, some sort of error detection-error correction coding must be added to prevent compressed data points from becoming altered or lost. This additional coding results in added redundancy to the data. A natural question is, "Why compress data in order to remove redundancy and then add some redundancy?" The answer is, "All the added redundancy is in an efficient code designed to find errors." Controllable redundancy is much more desirable than uncontrollable redundancy.

APPLICATIONS OF DATA COMPRESSION TECHNIQUES

Two desirable by-products of using pre-transmission data compression techniques are:

1. Reduction in the amount of data to be transmitted.
2. Reduction in data volume

The use of data compression techniques in ground-to-ground communication lines would enable greater capacity at present rental rates. Lower capacity links such as teletype and dataphone might be capable of carrying present microwave link capacities. In the spacecraft a gain in signal-to-noise ratio, weight, or power, depending on system payoffs, could be realized.

Preprocessing data compression would reduce the input-output burdens of the data processing facilities enabling a faster turn around time. The specialized format of the compressed data would also contribute to faster decommutation times. Magnetic tape inventories may be significantly reduced allowing savings in tape storage facilities as well as in inventory.

Figure 4 shows the data handling loop from experimenter to satellite to data processing to experimenter. Places where data compression may be used are

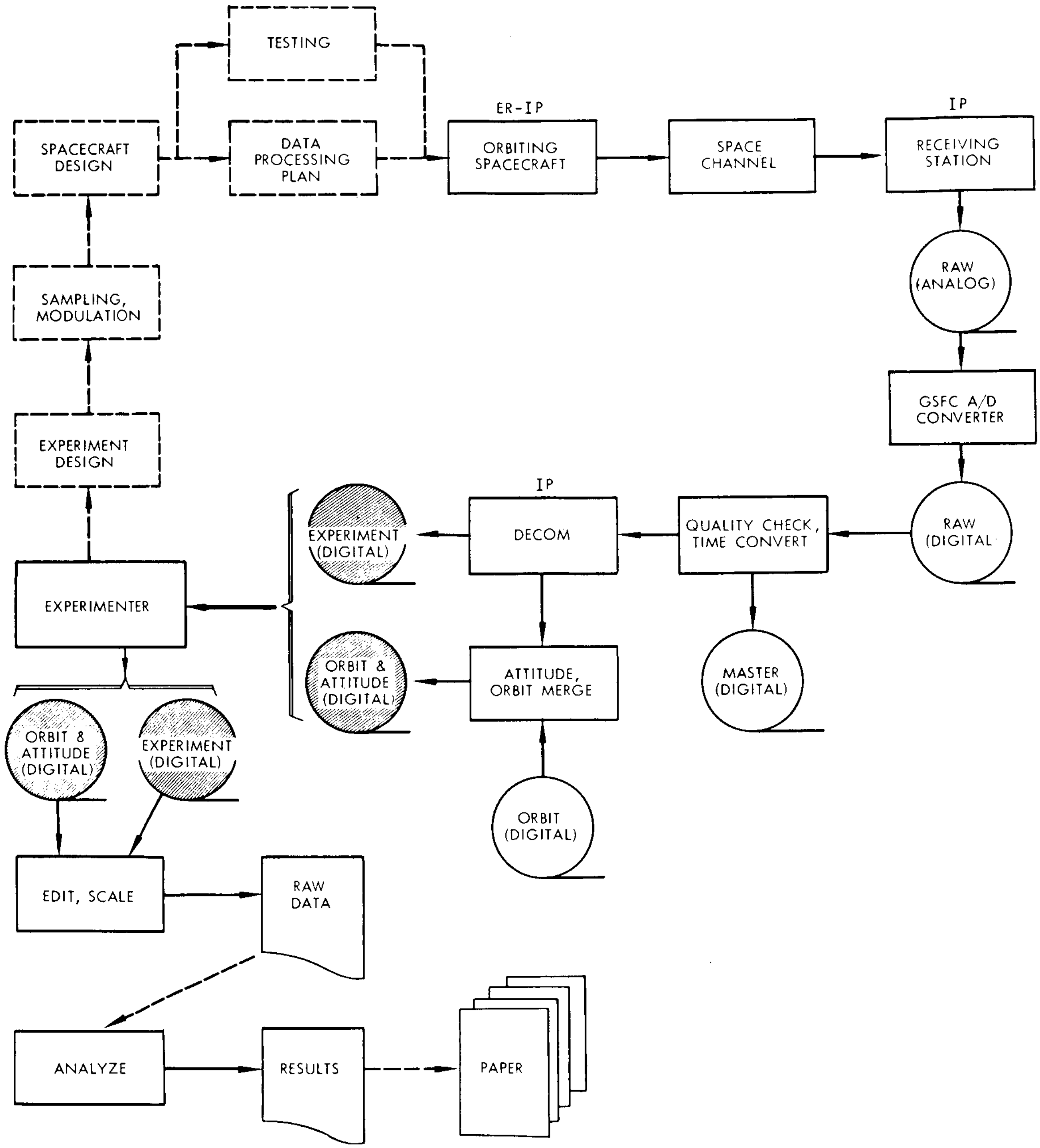


Figure 4–GSFC Space Data Handling

indicated by ER or IP next to the function where data compression could be incorporated.

In order to gain maximum benefit from data compression the two types, Entropy Reduction and Information Preserving, must be closely matched so that they complement each other. Efficiency is lost when ER and IP data compression are implemented separately in the same system, because of the complex waveform associated with most ER compression schemes.

At the present time only a few IP compression facilities have been implemented and these are all ground systems. An IP compressor of the zero-order predictor type has been built for use on the Saturn I Launch Vehicle, but has not been flown yet. The IP compression has also been considered for use on scientific spacecraft. As the communication loads increase one can assume that IP data compression will come into wide use.

Entropy Reduction data compression has been in use for quite some time in the various spacecraft experiments. The increasing demands for more power, less weight, less bandwidth, etc., are forcing more and more experimenters to come up with designs which will send less data and more information for less power.

Use of data compression could be the answer for a faster, more adaptable telemetry system and data handling operation.

BIBLIOGRAPHY

1. Blasbalg, H., and Van Blerkom, R. Message Compression. Northeast Electronics Research and Engineering Meeting, October 24, 1961.

2. Blasbalg, H., op. cit.

3. Weber, D., A Synopsis on Data Compression. National Telemetry Conference, 1965.

4. Final Report. "Data Compression and Time Resolution Study" NAS5-9729.

5. Blasbalg, H., op. cit.

6. Simpson, R. Buffer Control in Data Compression Systems for Non-Stationary Data. National Telemetry Conference, 1964.

7. Weber, D., op. cit.

8. Weber, D., op. cit.

9. Schwartz, J. Data Processing in Scientific Space Probes. Yale University, September 1963.

APPENDIX A

Basic Data Compression Algorithms

1. Zero-Order Floating-Aperture Predictor

This algorithm predicts that the following samples will be within a specified tolerance of the first sample. If this prediction is true, the data sample is considered redundant and is not transmitted. The first sample which falls outside the tolerance level is considered non-redundant, is transmitted, and becomes the new reference for subsequent predictions. Tolerances will normally be different for each channel, but will be variable by command.

2. Zero-Order Interpolator

In this algorithm the tolerance is set about the first data point as was done in the Zero-Order Predictor. When the next point is examined the tolerance is set up about it. The windows of the tolerances of the two points are checked for an overlap. If the windows do overlap, a new window upper bound is found by taking the minimum of the upper bound of the old window verses the new point plus the half-tolerance. A new lower bound is found by similarly taking the maximum of the lower bound verses the new point minus the half-tolerance. The tolerance is set up about the third point, the window formed, checked for overlap against the last window generated, a new window formed, etc. When the windows fail to overlap, an average value is found by taking the average of the upper and lower bounds. This value is then transmitted and a new run started.

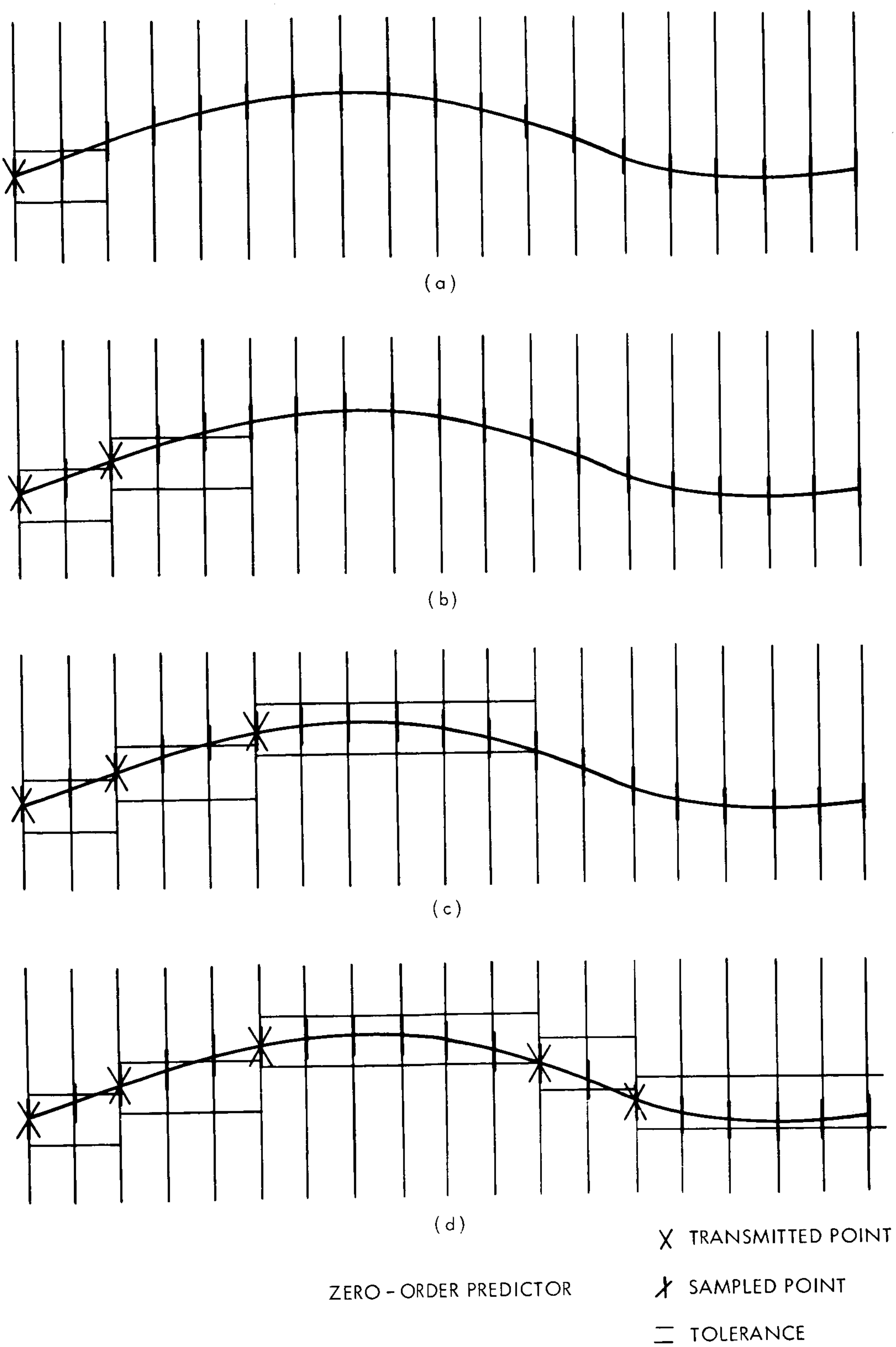
3. First-Order Predictor

In this algorithm the first point is transmitted. The third point is predicted to be on the line connecting the first and second points, plus or minus the given tolerance. If the point is in tolerance the next point is predicted on the same line, etc. If the new point was outside the tolerance limits, the previous predicted sample is transmitted and a new run started using the next value.

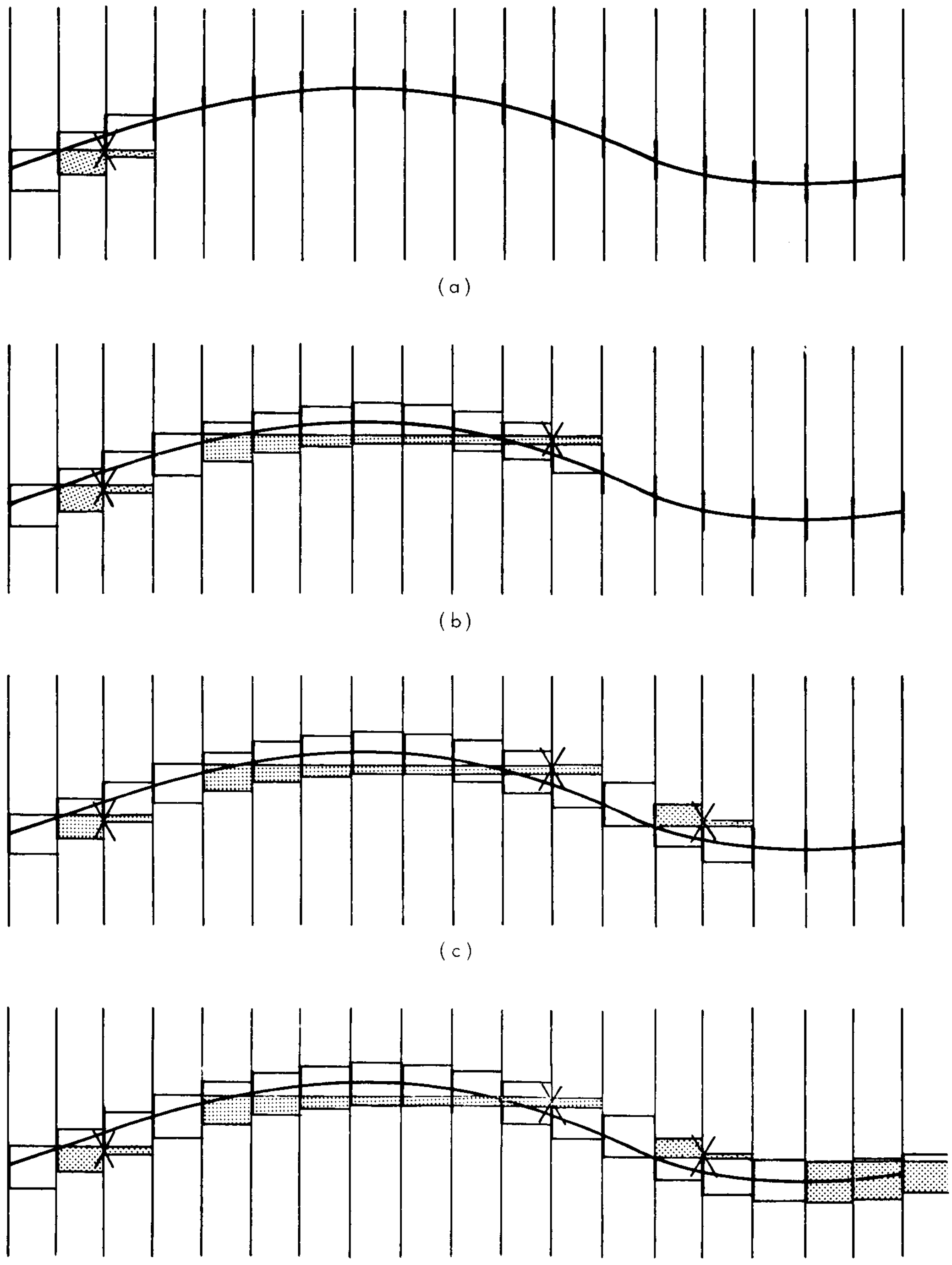
4. <u>First-Order Interpolator</u>

In this algorithm the first point is transmitted. The given tolerance is placed about the second point and lines drawn from the first point through the limits of the tolerance. If the third point is within the "fan" thus formed, a new "fan" is started by drawing lines between the first point and the tolerance limits placed about the third point. The new "fan" will be the intersection[1] of the two "fans." Subsequent "fans" are formed until a point does not fall within the "fan." A mean value of the last tolerance spread is transmitted and a new sequence begun using the last actual point.

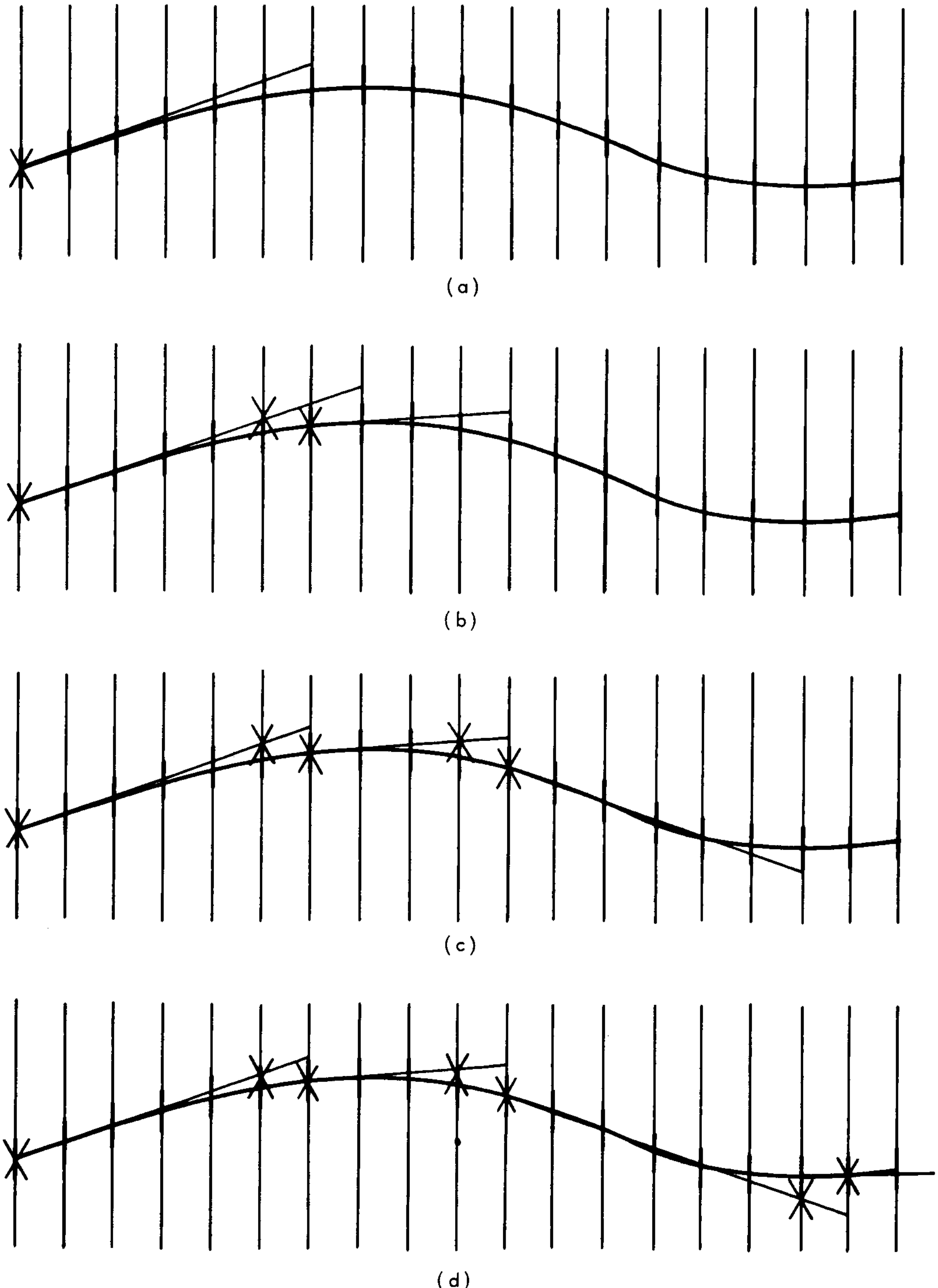
[1]Intersection here refers to the intersection of two sets, fan 1 and fan 2.



ZERO - ORDER PREDICTOR



ZERO - ORDER INTERPOLATOR

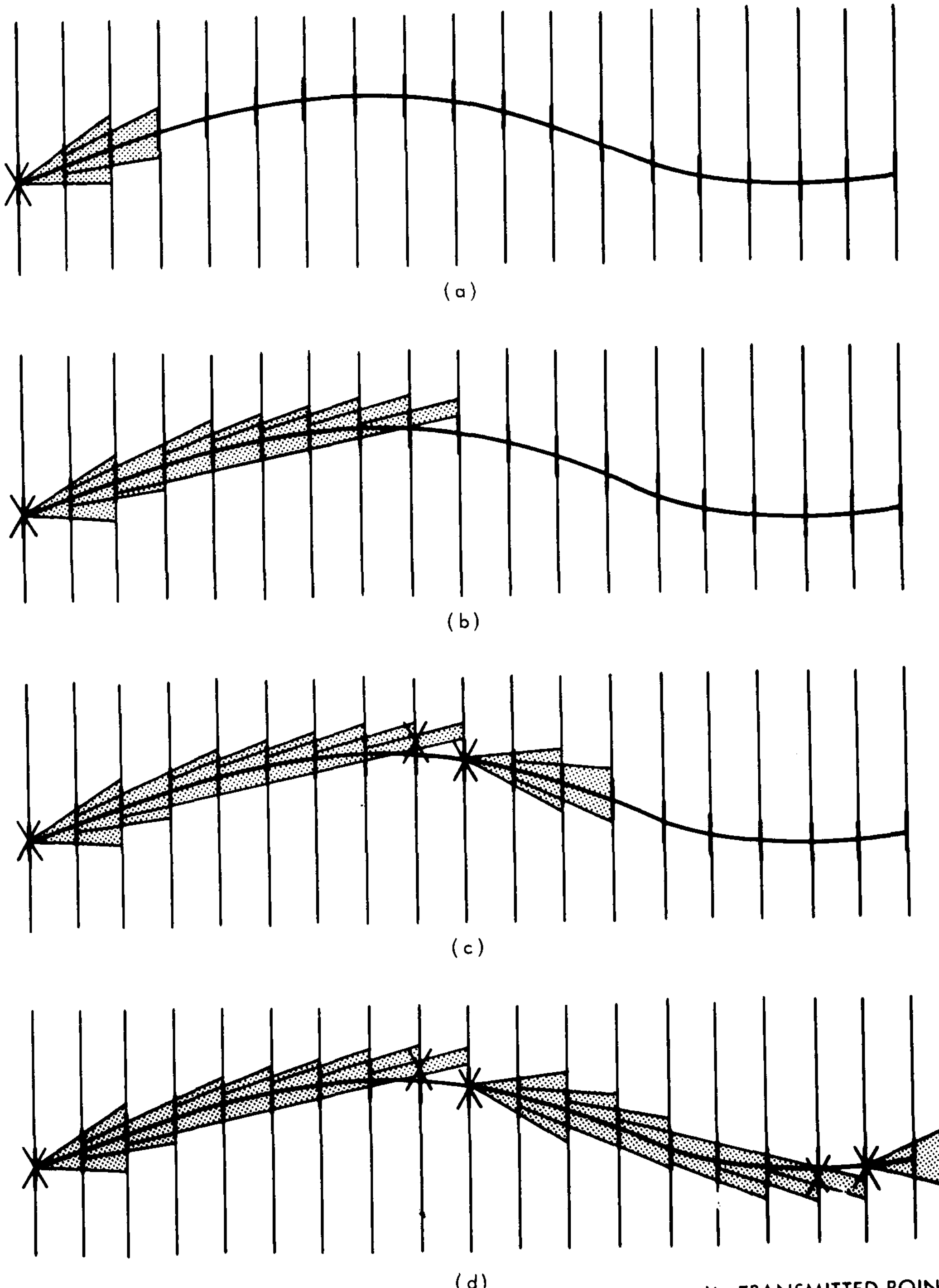


X TRANSMITTED POINT

FIRST - ORDER PREDICTOR

✗ SAMPLED POINT

= TOLERANCE



X TRANSMITTED POINT

FIRST - ORDER INTERPOLATOR

SAMPLED POINT

TOLERANCE

APPENDIX B

Proposed Data Compression Terms

1. Data Compression. The term which describes any method of decreasing the number of bits required to describe the information from a data source prior to transmission, processing, storage, etc., operations on the data.

2. Entropy Reducing Data Compression. A class of irreversible operations on a data source which reduce the original signal to a code that enables the necessary information to be encoded using fewer bits.

3. Information Preserving Data Compression. A class of reversible operations which produce a code capable of reproducing the original data to within a specified error using fewer bits than the conventional telemetry system.

4. Tolerance. The range of deviation in which data variations are not significant due to noise or instrumentation parameters. This value is used to determine the size of the "window" or aperture in IP data compression algorithms.

5. Compression Algorithm. Techniques used to implement information preserving data compression of the polynomial curve fit class.

6. Bit Compression Ratio. A measure of compression system efficiency. Equal to the number of bits input to the compression system divided by the number of bits transmitted by the telemetry system.

7. Word Compression Ratio. A measure of data compression efficiency. Equal to the number of words input to the data compressor divided by the number of words output from the compressor.

8. Data Compression Buffer. A temporary storage device which enables the asynchronous output from the data compression algorithm to be transmitted synchronously.

9. Adaptive Aperture Control. A method of controlling the data compression buffer fullness by setting the tolerance or aperture window to vary the data point rejection level of the data compression algorithm, so that the input rate to the buffer will increase or decrease.

10. Adaptive Filtering. A type of buffer control which oversamples the data near the frequency of the system noise. An average is taken of N points to smooth the data. The average value is transmitted. N can be varied in order to change data rates.

11. Adaptive Sampling. A type of buffer control which varies sampling rates at the commutator according to the data activity in order to vary the input to the data compressor.

12. Data Activity. A measure of the rate of change of the significant data values. The higher the frequency of change the higher the data activity.