JPL Publication 87-19

Noiseless Coding for the Magnetometer

Robert F. Rice **Jun-Ji Lee**

June 15, 1987

NASA
National Aeronautics and
Space Administration

Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California

JPL Publication 87-19

Noiseless Coding for the Magnetometer

Robert F. Rice **Jun-Ji Lee**

June 15, 1987

NASA
National Aeronautics and
Space Administration

Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California

The research described in this publication was carried out by the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

ABSTRACT

Future unmanned space missions will continue to seek a full understanding of magnetic fields throughout the solar system. Severely constrained data rates during certain portions of these missions could limit the possible science return. This publication investigates the application of universal noiseless coding techniques to more efficiently represent magnetometer data without any loss in data integrity. Performance results indicated that compression factors of 2:1 to 6:1 can be expected. Feasibility for general deep space applications was demonstrated by implementing a microprocessor breadboard coder/decoder using the Intel 8086 processor.

The Comet Rendezvous Asteroid Flyby mission will incorporate these techniques in a buffer feedback, rate-controlled configuration. The characteristics of this system are discussed.

ACKNOWLEDGMENT

The authors wish to thank: Ron Draper for his programmatic support of this work; Jim Slavin, Ed Smith, Bruce Tsurutani, Roy Marquedant and Chris Russell for their technical advice; and Bruce Parham for demonstrating the feasibility of implementing these results in microprocessor form.

CONTENTS

Figures

Tables

I. INTRODUCTION

A full understanding of the magnetic fields throughout the solar system has been an important scientific objective since the inception of the deep space program[1]-[3]. Both the Mars Observer (MO) and the Comet Rendezvous Asteroid Flyby (CRAF) missions are expected to fly sophisticated magnetometers to continue this investigation.

The purpose of this publication is to provide a preliminary understanding of the potential advantages that may be derived from applying certain well-defined data compression techniques to magnetometer instruments. The work described herein was primarily performed in 1985. The results were incorporated into the instrument proposal which was eventually selected for the CRAF mission[4].

This publication is the second such study for science instruments. The results of a similar study for the Gamma Ray Spectrometer are described in Ref. 5. The latter results are currently being implemented for the MO mission.

EXPERIMENT DATA GATHERING MODEL

Figure 1 provides a simple interpretation of the magnetometer data gathering process. The magnetometer samples the three-dimensional magnetic field every τ seconds along the spacecraft path of motion.

Based on the characteristics of the ISEE-C Vector Helium Magnetometer described in Ref. 2, the measurement vector resulting at time t_i can be denoted by

$$\tilde{v}_i = r(i)\ x(i)\ y(i)\ z(i) \tag{1}$$

where the x(i), y(i), z(i) are m-bit representations of the three orthogonal components resulting from a linear quantizer operating over a **range** of signal values specified by the identifier r(i).

The instrument described in Ref. 2 has the ability to manually or automatically switch between eight dynamic ranges, whereas the actual instrument built for MO or CRAF may have fewer ranges. In order to include the existence of r(i) within the overall coding problem, we let b_r be the required length of r(i) in bits where

$$1 \leq \beta_r \leq 3 \tag{2}$$

depending on the number of instrument dynamic ranges.

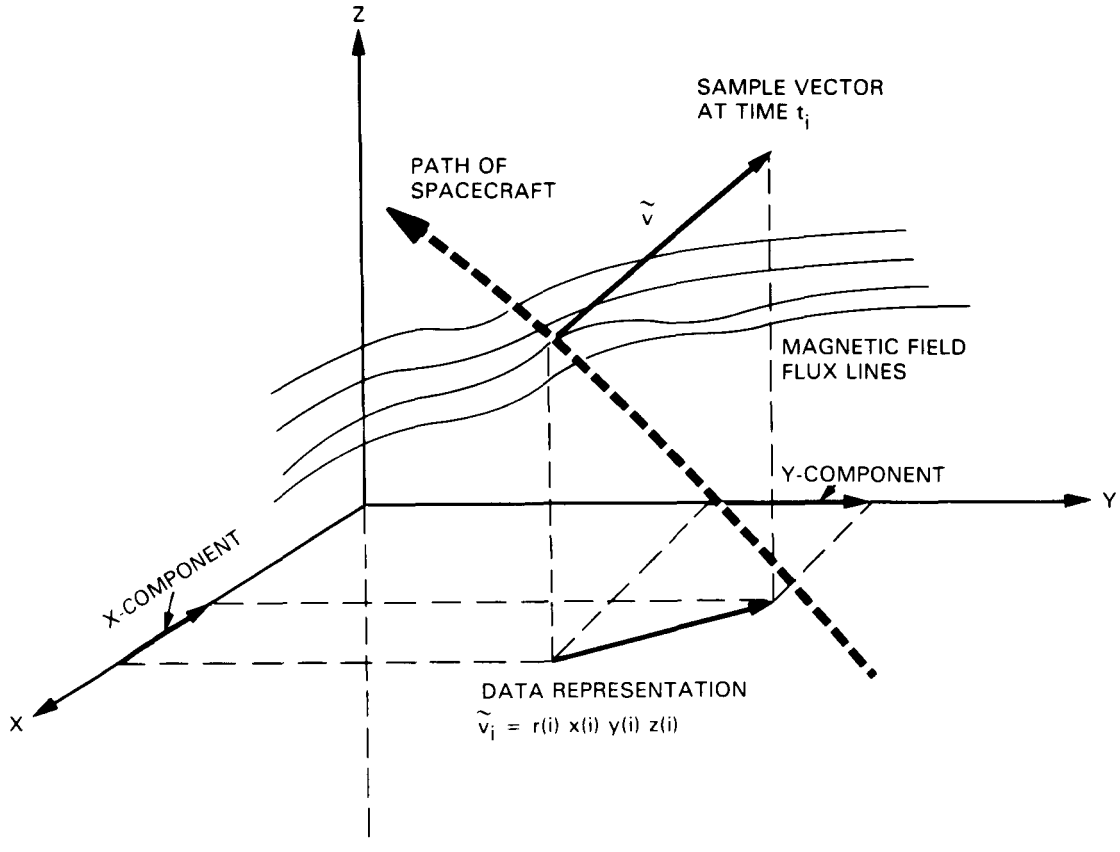

Fig. 1. Magnetometer Measurements.

Then the baseline representation of the sampled magnetic field requires

$$3m + \beta_r \text{ bits/measurement vector (b/v)} \tag{3}$$

The data rate resulting from continuous sampling is then

$$R_l\,(m, \beta_r, \tau) = \frac{3m + \beta_r}{\tau} \text{ b/s} \tag{4}$$

Desires and Constraints

When instrument parameters have been sized to provide a basic maximum data-gathering capability compatible with the science objectives of both MO and CRAF, the following parameters result:

$$m \leq 12 \text{ b/component} \tag{5}$$

and

$$1/\tau \leq 20 \text{ vectors/s (v/s)} \tag{6}$$

Using Eq. (4) with $\beta_r = 3$ we get a maximum instrument data rate of

$$R_I^{max} \approx 800 \text{ b/s} \tag{7}$$

Unfortunately, data rate constraints placed on the magnetometer instrument will be considerably lower than this. As a consequence, the parameters m and τ would need to be altered significantly to meet this constraint. These basic trade-offs are graphed in Fig. 2. Certainly, such adjustments would likely reduce the science value of the returned measurements.

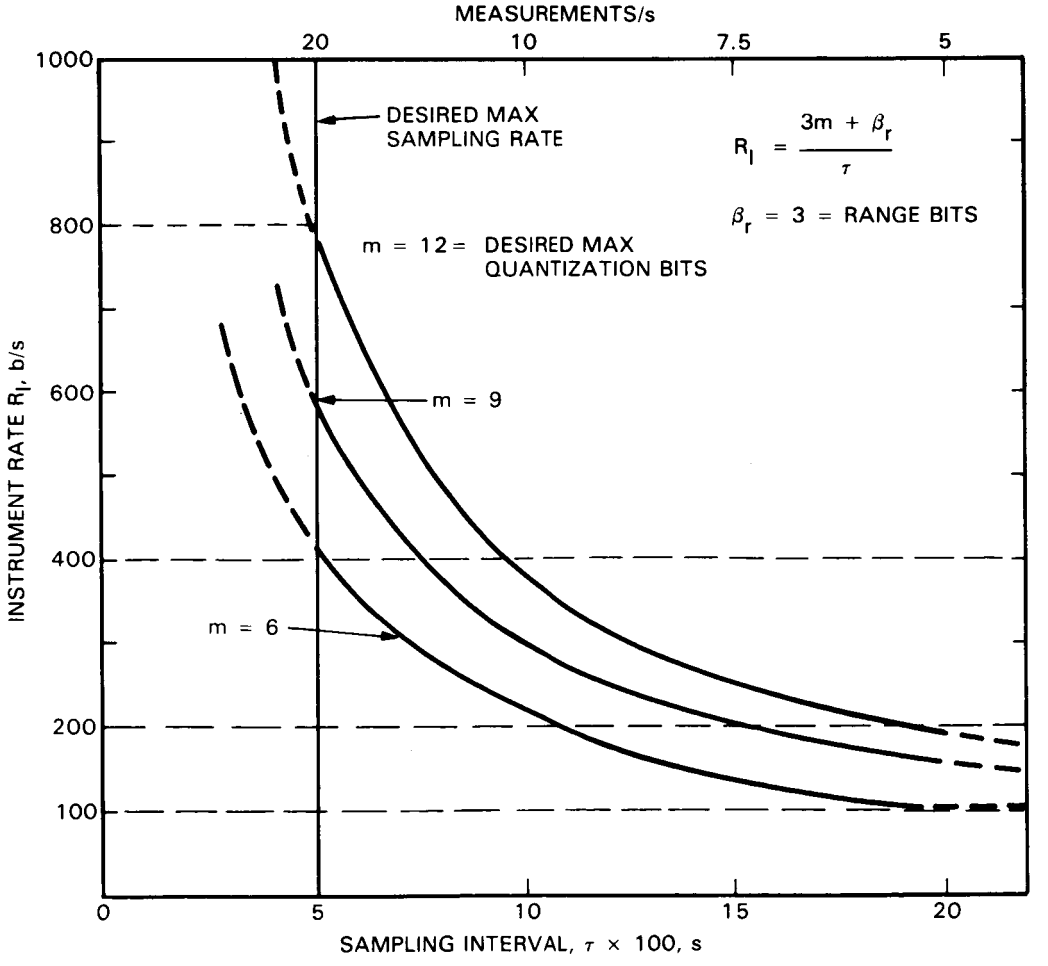

Fig. 2. Baseline Instrument Data Rates, R_I Versus m and τ.

The remainder of this publication investigates data compression as a means of obtaining such required reductions in data rate with higher data integrity (i.e., less modifications to m and τ).

DATA COMPRESSION STRUCTURE

The techniques previously applied to image data[6]-[9] and the Gamma Ray Spectrometer[5] are equally suitable here. Figure 3 shows the general data compression structure.

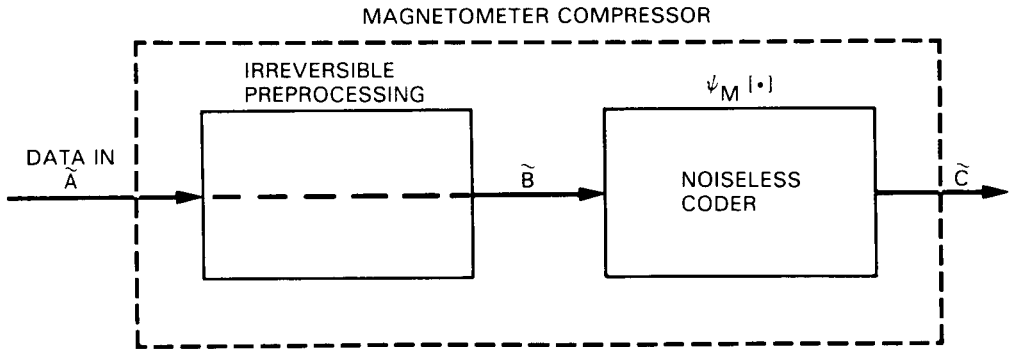

Fig. 3. Overall Data Compression Structure.

As shown, data compression generally includes two steps. The first, called "irreversible preprocessing," consists of **modification** to the input data which cannot be undone by a decompressor. Such irreversible preprocessing could include such things as modifications to sample quantization or effective sampling rate. Irreversible preprocessing can provide reductions in rate, but its consequence is a **reduction in the fidelity** of reconstructed data.

On the other hand, the second step in Fig. 3, called "noiseless coding," can achieve a reduction in rate **without reducing the fidelity of data encoded**. In Fig. 3, this means that the output of the noiseless coder, $\tilde{C}$, can be "decompressed" to reconstruct $\tilde{B}$ exactly (but not necessarily $\tilde{A}$).

The output of the overall compression structure in Fig. 3 will therefore **reconstruct** with a quality corresponding to $\tilde{B}$, but operate at a rate corresponding to $\tilde{C}$.

Noiseless Compressor

If we constrain

$$\tilde{B} = \tilde{A} \tag{9}$$

we obtain a magnetometer compressor which results in exact reconstruction, that is, a "noiseless" magnetometer compressor. We will **initially** restrict attention to this form as indicated by the dashed line in Fig. 3. Any changes in m and τ can be considered as changes to the instrument output parameters resulting in input $\tilde{A}$.

Once we have defined and analyzed such a noiseless coder, we will return to the subject of irreversible preprocessing. An efficient noiseless coder will produce data rates which vary with data activity. For applications such as CRAF, which have fixed instrument data rate allocations, this necessitates some form of "rate control." We will describe a buffer feedback system for CRAF which accomplishes this objective by making fine adjustments to the effective sampling rate (i.e., varying τ)[4].

II. NOISELESS MAGNETOMETER COMPRESSOR

The noiseless coding required here, as for the Gamma Ray Spectrometer[5], is a direct application of techniques described in Refs. 6 through 9. Whenever possible we will use the notations and definitions already described in these reports.

PRELIMINARIES

Notation

Key notational quirks are worth noting before proceeding:

1) We will generally place a tilde ($\sim$) over a symbol to indicate a **sequence** of bits or other symbols;

2) An asterisk (*) will be used to indicate concatenation of symbols or sequences when we need some additional emphasis;

3) If $\widetilde{Z}$ is a sequence, we use $\pounds(\widetilde{Z})$ to indicate its **length** in bits; and

4) We subscript and superscript $\psi[\bullet]$ to indicate a particular noiseless code operator. Specifically, if $\widetilde{Z}$ is an input sequence to some code operator $\psi_a[\bullet]$, then $\psi_a[\widetilde{Z}]$ is the resulting coded sequence **and**, given $\psi_a[\widetilde{Z}]$, the original sequence $\widetilde{Z}$ can be recovered precisely.

General Structure

The noiseless coder in Fig. 3 generally splits into separate operations as indicated in Fig. 4.

In addition to formatting incoming data, the function of reversible preprocessing is essentially to predict sample values and then map the prediction error into the integers such that the most likely occurring integers are the smaller ones[6],[7]. The better this last objective is achieved, the better the variable length coder that follows will work. That is, a better predictor will lead to lower average coded rates. The process is reversible in that the original data can be recovered from the mapped integers.

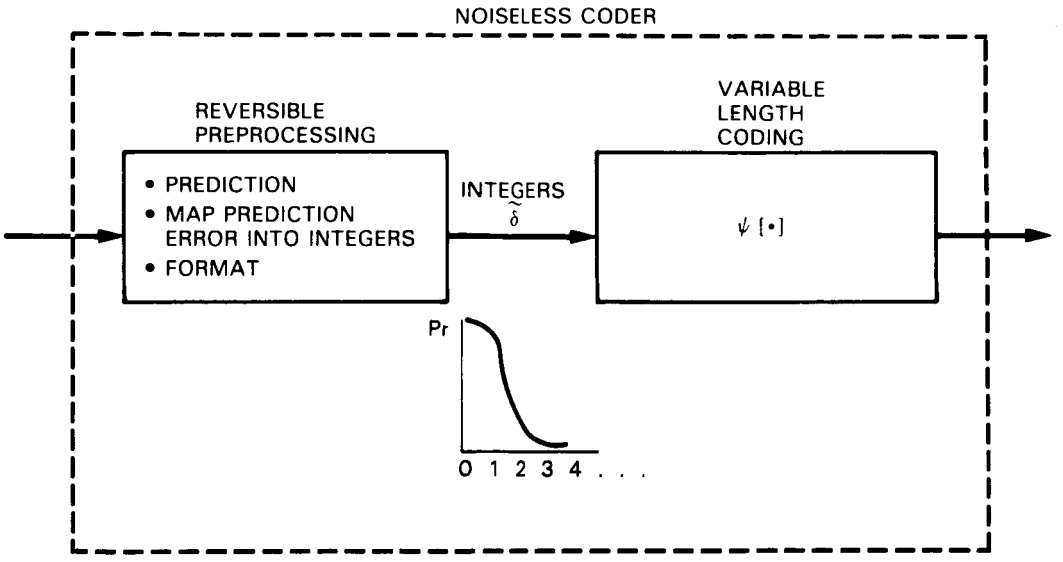

Fig. 4. Overall Noiseless Coding Structure.

The step of reversible preprocessing is generally source dependent, whereas the step of variable length coding can generally be made source independent. Its implementation usually consists of selecting the appropriate algorithms from Refs. 6 and 7, which efficiently code integers over the range of data activity expected for the given application.

SEPARATING THE DATA SOURCE

Magnetometer data as defined in Eq. (1) really consists of four separate data sources which need to be dealt with individually. It will be convenient later to have isolated this step of preprocessing, and hence the following discussion will lead to an intermediate definition of a magnetometer compressor.

Referring to Fig. 1, let

$$\tilde{V} = v_1 * v_2 * \ldots * v_J \tag{10}$$

be a **block** of J consecutive measurement vectors starting at time t_i. That is, using Eq. (1)

$$\tilde{V} = r(1)x(1)y(1)z(1) * r(2)x(2)y(2)z(2) \ldots . \\ \ldots . * r(J)x(J)y(J)z(J) \tag{11}$$

Now collect all the range and the x, y, z components into individual blocks

$$\begin{aligned} \tilde{R} &= r(1)\ r(2)\ \ldots\ r(J) \\ \tilde{X} &= x(1)\ x(2)\ \ldots\ x(J) \\ \tilde{Y} &= y(1)\ y(2)\ \ldots\ y(J) \\ \tilde{Z} &= z(1)\ z(2)\ \ldots\ z(J) \end{aligned} \tag{12}$$

and concatenate them into the new sequence

$$\tilde{V}' = \tilde{R} * \tilde{X} * \tilde{Y} * \tilde{Z} \tag{13}$$

Clearly these rearrangements are reversible; $\tilde{V}$ can be recovered from $\tilde{V}'$. The steps are illustrated in Fig. 5.

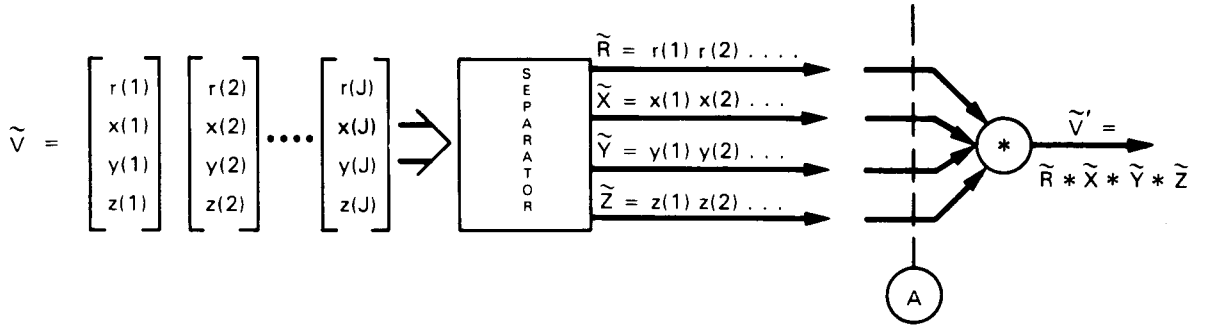

Fig. 5. Separating Into Multiple Data Sources.

Note that $\tilde{V}'$ has the same form as the original single measurement vector $\tilde{v}_i$ in Eq. (1), except that now each segment is a J sample sequence of the same kind of data.

We can now treat local segments of the data sources for range and x, y, z by separately coding the blocks $\tilde{R}$, $\tilde{X}$, $\tilde{Y}$ and $\tilde{Z}$ at point Ⓐ in Fig. 5. Using the code operator notation of Refs. 6 and 7, let

$$\psi_R[\bullet],\ \psi_x[\bullet],\ \psi_y[\bullet],\ \psi_z[\bullet] \tag{14}$$

represent the noiseless code operators to be applied to $\tilde{R}$, $\tilde{X}$, $\tilde{Y}$ and $\tilde{Z}$, respectively.

From Fig. 5 and Eq. (14), the magnetometer noiseless coder (denoted $\psi_M[\bullet]$) takes the form shown in Fig. 6. We need only identify the operations of the various $\psi[\bullet]$s.

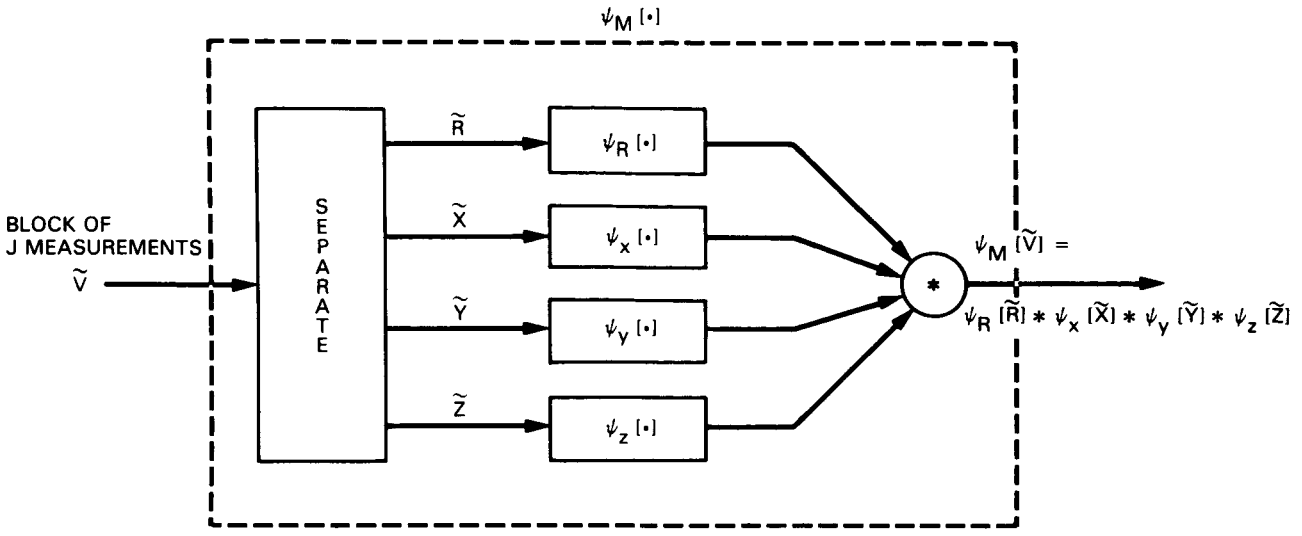

Fig. 6. Magnetometer Noiseless Code Operator, $\psi_M[\cdot]$.

The output of the magnetometer code operator as shown in Fig. 6 is defined by

$$\psi_M[\tilde{V}] = \psi_R[\tilde{R}] * \psi_x[\tilde{X}] * \psi_y[\tilde{Y}] * \psi_z[\tilde{Z}] \tag{15}$$

and, since each of the individual operations are reversible (by definition), $\tilde{R}$, $\tilde{X}$, $\tilde{Y}$ and $\tilde{Z}$ can be recovered from $\psi_M[\tilde{V}]$, from which $\tilde{V}$ itself can be retrieved.

CODING $\tilde{X}$, $\tilde{Y}$ AND $\tilde{Z}$

Although $\tilde{X}$, $\tilde{Y}$ and $\tilde{Z}$ need to be treated separately, they can each use the same coding process, which at this point is identical to an image compressor as defined in Ref. 7. For convenience, let

$$\psi_\lambda[\cdot] \tag{16}$$

be the general purpose code operator which applies to $\tilde{X}$, $\tilde{Y}$ and $\tilde{Z}$ so that

$$\psi_x[\cdot] \equiv \psi_y[\cdot] \equiv \psi_z[\cdot] = \psi_\lambda[\cdot] \tag{17}$$

Now let $\tilde{\Lambda}$ be an input sequence to $\psi_\lambda[\cdot]$ (i.e., $\tilde{X}$, $\tilde{Y}$ or $\tilde{Z}$), and let $\alpha(j)$ be the j^{th} element of $\tilde{\Lambda}$ so that

$$\tilde{\Lambda} = \alpha(1)\ \alpha(2)\ \ldots\ \alpha(J) \tag{18}$$

Now if we use $\alpha(j-1)$ as the **prediction** of α_j we get the j^{th} prediction error†

$$\Delta_j = \alpha(j) - \alpha(j-1) \tag{19}$$

and a J sample block of prediction errors

$$\tilde{\Delta} = \Delta_1 \, \Delta_2 \, . \, . \, . \, \Delta_J \tag{20}$$

If we then map each of the individual errors, Δ_j, into the integers we get

$$\tilde{\delta} = \delta_1 \, \delta_2 \, . \, . \, . \, \delta_J \tag{21}$$

where

$$\Delta_j \rightarrow \delta_j \tag{22}$$

according to the relationship in Table 1.

Table 1. Basic Mapping of Δ Into the Integers δ.

PREDICTION ERROR Δ	INTEGER δ
0	0
+1	1
−1	2
+2	3
−2	4
+3	5
−3	6
•	•
•	•
•	•

†The last sample of the previous Λ block can be used for $\alpha(0)$. If a previous block does not exist, $\alpha(0)$ can be a constant.

The latter steps are precisely the same steps shown in more detail in Ref. 7 and constitute the "reversible preprocessing" of code operator $\psi_\lambda[\cdot]$.† The remaining step is provided by a variable length coder which we will denote here as

$$\psi_\ell\,[\cdot] \tag{23}$$

and which can be chosen from numerous alternatives in Refs. 6 and 7.

The code structure for handling $\tilde{X}$, $\tilde{Y}$ or $\tilde{Z}$ through $\psi_\lambda[\cdot]$ is shown in Fig. 7.

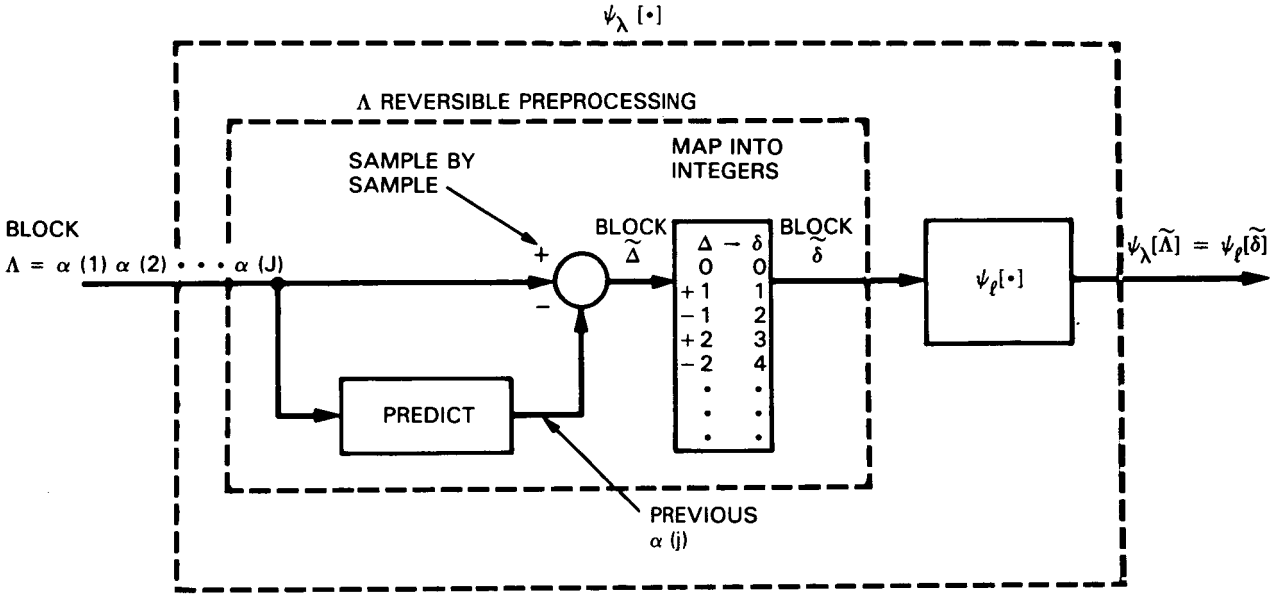

Fig. 7. Code Operator $\psi_\lambda[\cdot]$ for $\tilde{X}$, $\tilde{Y}$, or $\tilde{Z}$.

Modifications for Frequent Range Changes

Note that when a range change occurs, the samples $\alpha(j-1)$ and $\alpha(j)$ are generated from different dynamic ranges. The error Δ_j in Eq. (19) could be quite large even if the corresponding analog sample values were the same. The net effect of a large Δ_j (a spike) would be an increase in code rate for the block in which it occurred. All our test data showed a negligible effect on long-known average coder performance.

However, in the event that more frequent range changes became the norm, prediction errors could be significantly reduced by mapping the prediction value $\alpha(j-1)$ from its range into the range of the next sample. Further, Ref. 7 provides supplemental techniques for improving efficiency in data blocks containing spikes (e.g., occasional large errors).

CODER FOR RANGE

The same principles apply to range data, although its characteristics are quite different than that of the x, y and z components. Changes in range are quite infrequent, so that one would expect that the rate required to communicate these changes should be considerably smaller than the 1 to 3 b/sample assumed in the basic representation of Eq. (2). This is in fact the case.

Our primary test data had eight ranges, so we will make that assumption here although actual CRAF or MO instruments will probably have fewer ranges. The code operator defined below for range data can be viewed as simply an expedient definition which, from our limited test set, achieves average rates which are negligible compared to that of $\tilde{X}$, $\tilde{Y}$ and $\tilde{Z}$. We have also built in some basic protection against unexpected conditions. Additional sophistication to improve efficiency would certainly be unwarranted.

Let

$$\zeta(i) = \begin{cases} 0 \text{ if } r(i) = r(i-1) \\ 1 \text{ otherwise} \end{cases} \tag{24}$$

and

$$\omega(i) = \begin{cases} \phi \text{ if } \zeta(i) = 0 \\ r(i) \text{ otherwise} \end{cases} \tag{25}$$

where $\phi \equiv$ non-existent, null or "no data" (see below). Then define code operator $\psi_R^1[\cdot]$ by

$$\psi_R^1[\tilde{R}] = \zeta(1)\ \omega(1)\ \zeta(2)\ \omega(2) \cdots \zeta(J)\ \omega(J) \tag{26}$$

In words, the $\zeta(j)$ are 0's and 1's indicating if a range has changed, and if a change has occurred ($\zeta = 1$) that ζ is followed by a three-bit (in general β_r) specification of the new range. If a particular ζ equals zero, a new range value is not needed ($\omega = \phi$) and that ζ is instead followed by the **next** ζ.

Code operator $\psi_R^1[\cdot]$ has two areas of inadequate performance. In the first, if all the range samples changed, $\psi_R^1[\cdot]$ would require four b/sample. Although unlikely, it is worthwhile to protect against this transient situation.

First recall that by the notation of Refs. 6 and 7, and earlier discussions, if $\tilde{K}$ is a sequence

$$£(\tilde{K}) \tag{27}$$

---

†Ref. 7 also describes a slightly more efficient mapping algorithm.

means the length of that sequence in bits. Now define

$$e = \begin{cases} 1 \text{ if } £\, \psi_R^1[\tilde{R}] > 3J \\ 0 \text{ otherwise} \end{cases} \tag{28}$$

and let

$$\psi_R^2[\tilde{R}] \doteq \begin{cases} 0 * \psi_R^1[\tilde{R}] & \text{if } e = 0 \\ 1 * \tilde{R} & \text{if } e = 1 \end{cases} \tag{29}$$

A close look at this operator reveals that $\psi_R^2[\tilde{R}]$ only uses $\psi_R^1[\bullet]$ to code $\tilde{R}$ when it actually provides a gain over no coding at all. The worst-case code rate for $\psi_R^2[\bullet]$ is (3 + 1/J) b/sample.

The second deficiency in $\psi_R^1[\bullet]$ (and the one remaining in $\psi_R^2[\bullet]$) is that its minimum rate of one b/sample is too high. This occurs in the very likely event (by our test data) that no range changes are detected in a J sample block.

Now let

$$f = \begin{cases} 0 \text{ if no range changes} \\ \quad \text{in complete block} \\ 1 \text{ otherwise} \end{cases} \tag{30}$$

and define the operator $\psi_R^3[\bullet]$ by

$$\psi_R^3[\tilde{R}] = \begin{cases} 0 \text{ if } f = 0 \\ 1 * \psi_R^2[\tilde{R}] \text{ if } f = 1 \end{cases} \tag{31}$$

which by Eqs. (28) and (29) expands to

$$\psi_R^3[\tilde{R}] = \begin{cases} 0 \text{ if } f = 0 \text{ (no changes)} \\ 1 * \begin{cases} 0 * \psi_R^1[\tilde{R}] \text{ if } e = 0 \\ 1 * \tilde{R} \text{ if } e = 1 \end{cases} \end{cases} \tag{32}$$

Looking closely at Eq. (32) we see that the length of $\psi_R^3[\tilde{R}]$, $\pounds(\psi_R^3[\tilde{R}])$ is only one bit per block if there are no range changes (a high probability); and 2 + J + 3 (number of changes) up to a maximum of (2 + 3J) bits when the number of changes is greater than zero.

This is illustrated in Fig. 8 for a block size of J = 16.

OVERALL NOISELESS CODER

We can now expand the magnetometer noiseless code operator, $\psi_M[\cdot]$ in Fig. 6, into the more complete description in Fig. 9.

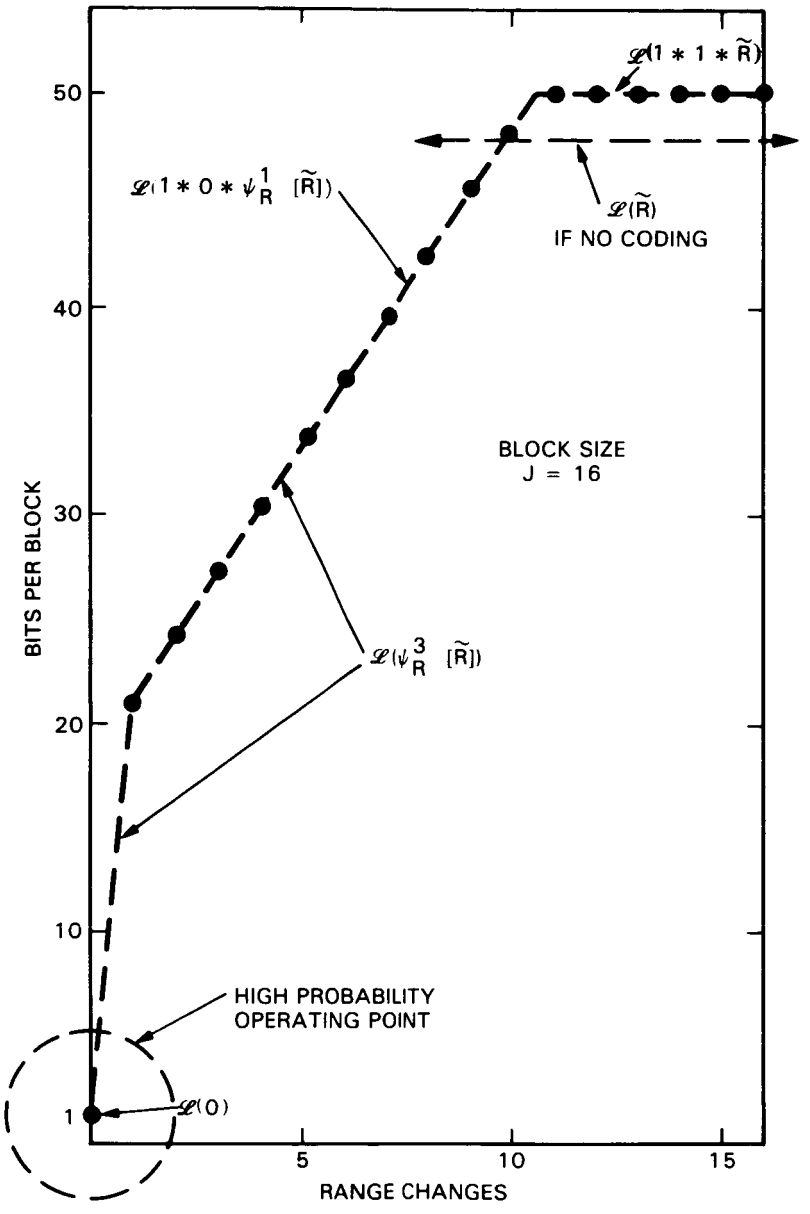

Fig. 8. Bits per Block $\mathcal{L}(\psi_R^3[\tilde{R}])$ Versus the Number of Range Changes.

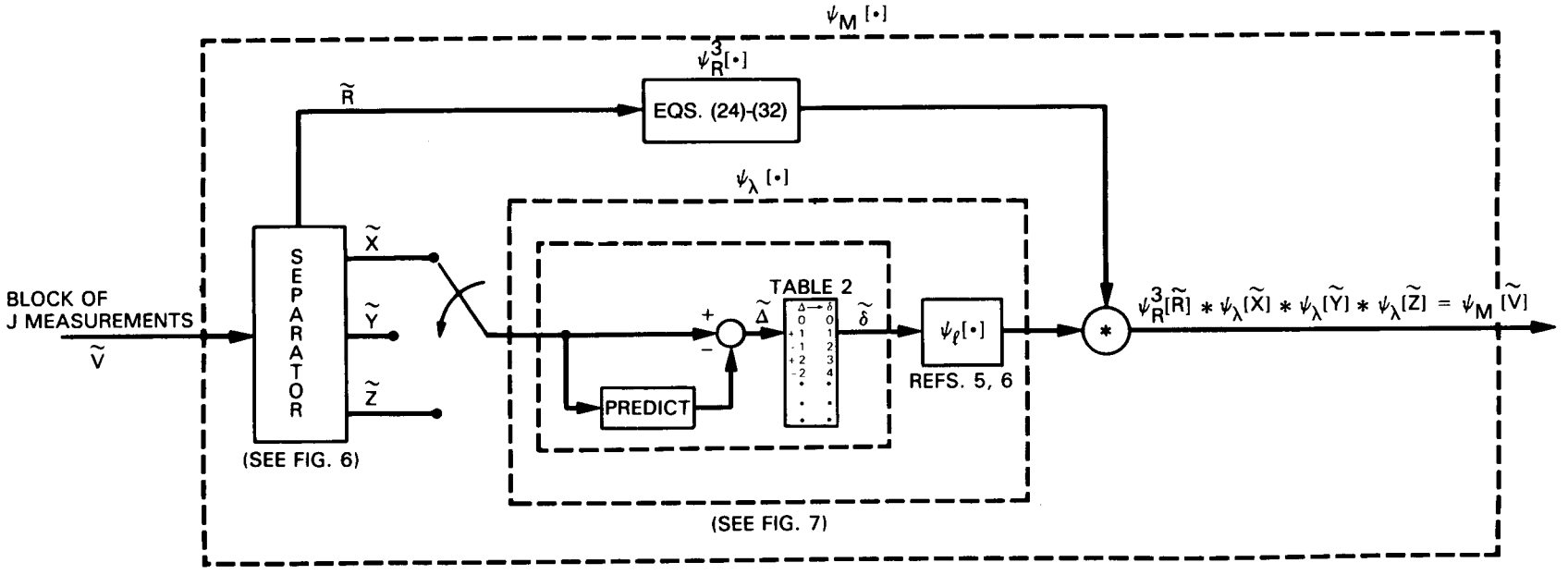

Fig. 9. Expanded Diagram of Magnetometer Noiseless Code Operator, $\psi_M[\cdot]$.

III. DATA COMPRESSION TESTS

ISEE-C DATA

The first magnetometer test set available to us was actual decalibrated data from the ISEE-C experiment described in Ref. 2. The data were represented by

$$m = 9 \text{ b/component} \tag{33}$$

which could be applied to eight dynamic ranges so that

$$\beta_r = 3 \tag{34}$$

The sampling rate was at

$$1/\tau = 6 \text{ vectors/s (v/s)} \tag{35}$$

which is roughly a factor of three slower than the desired 20 v/s, as noted in Eq. (6). While not a perfect match to the anticipated configurations for CRAF and MO, the test runs clearly provided a feel for first-order effects with some reasonable extrapolations possible.

Variable Length Coders

The only undefined element of the block diagram in Fig. 9 is the variable length coder, designated $\psi_\ell\,[\bullet]$. The three algorithms tested below are described in Refs. 6 and 7. These are:

A) An eight-option version of the "Fast Compressor" used at the recent Uranus encounter for Voyager imaging.

B) A four-option adaptive coder called the "Basic Compressor" or $\psi_4[\bullet]$.

C) An eight-option $\psi_{11}[\bullet]$ using $\psi_{i,j}[\bullet]$ (see, for example, Fig. 10 in Ref. 7). This version, while still simple and efficient, provides enough dynamic range to satisfy the requirements of most magnetometer instruments as well.

Since these algorithms are essentially source independent and thoroughly discussed in Refs. 6 and 7, we will not provide any more details here.

Performance Measurements

Average measured performance of some noiseless code operator will mean the average number of bits per (x, y or z) component required by that operator over some **specified measurement span** (i.e., sum the bits and divide by the number of measurements). We will look at such average performance from several different angles.

Average entropy calculated for some measurement span is a **practical guide** to the minimum possible rate one can expect to code the corresponding data and still remain noiseless. Under some conditions it is an absolute bound to performance [6],[7]. Here, entropy calculations ($-\Sigma\ p_i \log p_i$) were derived from distributions of prediction errors, $\tilde{\Delta}$ in Eq. (20) and range samples r(i).

Overall Average. We first look at the average code rates obtained over the complete test set by operator $\psi_M[\bullet]$ using the variable length code options A, B and C and $\psi_R^3[\bullet]$ as specified in Eqs. (24) through (32). The results are given in Table 2.

Table 2. Overall Average Performance of $\psi_M[\bullet]$.

ENTROPY (BITS/COMPONENT)	MEASURED PERFORMANCE (BITS/COMPONENT)		
	$\psi_\ell[\bullet]$ = OPTION A	$\psi_\ell[\bullet]$ = OPTION B	$\psi_\ell[\bullet]$ = OPTION C
3.56	3.88	3.45	3.42
*INCLUDES A RANGE ENTROPY OF $\approx$ 0.06 BITS PER RANGE SAMPLE	*INCLUDES AN AVERAGE CODE RATE OF 0.11 BITS PER RANGE SAMPLE USING $\psi_R^3[\bullet]$ in EQS. (24) TO (32).		

We see that Option C reduces the fixed rate of 10 b/component (30 b/v) to only 3.42 b/component **on the average.** This is an average compression factor of 2.92.

Variability. As noted earlier, the code rate actually achieved with an efficient noiseless coder will vary as the activity of the data (entropy) varies. The extent of this variation is important to buffering requirements and/or feedback control of quantization and sampling rate to meet data system rate limitations. We take a first order look at this phenomenon by comparing the entropies for $\tilde{X}$, $\tilde{Y}$ and $\tilde{Z}$ components and their composite averages taken over 10 test blocks of 44,000 vector samples each. The result is plotted in Fig. 10.

As the diagram shows, $\tilde{X}$, $\tilde{Y}$ and $\tilde{Z}$ activities follow the same trend in variation. In each case the observed swing between maximum and minimum entropies is about 2 b/component. The same fluctuations are thus observed on the composite average of the $\tilde{X}$, $\tilde{Y}$ and $\tilde{Z}$ graphs.

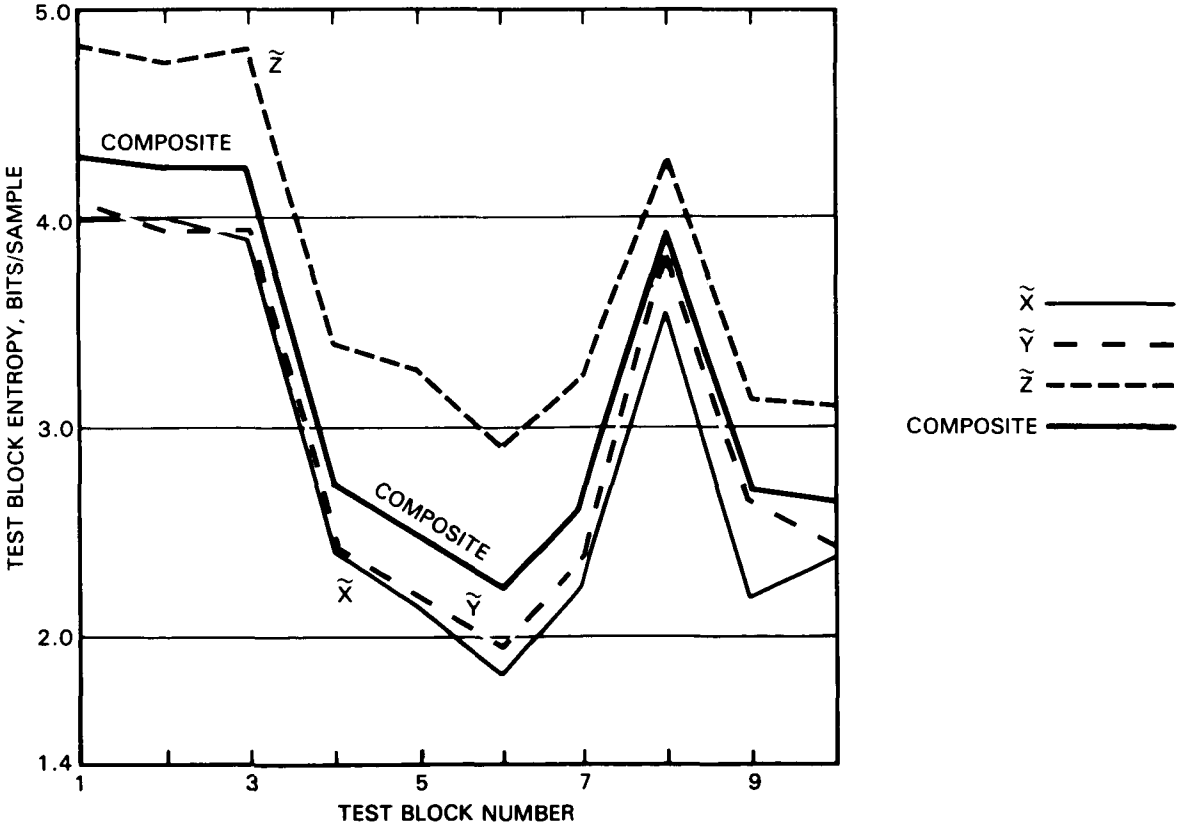

Fig. 10. Rate Variability Over Ten 44,000 Vector Test Blocks.

VENUS IONOSPHERE DATA[10]

Chris Russell of the University of California at Los Angeles provided us with several files of magnetometer data for the Venus ionosphere. The files contained data quantized to 12 b/component at sampling rates up to 24 v/s.

Applying the same noiseless coding techniques, we observed average compression factors of 3:1. Compression factors as high as 6:1 were observed over short stretches.

GIACOBINI-ZINNER COMET DATA[4]

Further investigations included the use of 1985 observations taken in the turbulent tail regions of Comet Giacobini-Zinner. These tests were used to evaluate a buffer-feedback, rate control system for the CRAF mission. The results are discussed in the next chapter.

IV. CRAF — A PRACTICAL APPLICATION

The noiseless coding techniques just discussed formed a part of the magnetometer instrument proposal which was accepted for inclusion in the CRAF science payload[4]. The following discussion looks at this practical application in more detail.

AN OPEN LOOP COMPRESSION SYSTEM

An expanded magnetometer compression system block diagram and some additional notation are shown in Fig. 11.

Raw Instrument Data Rates

The proposed instrument will generate 20 vectors/s (v/s) using 12-bit quantization on X, Y and Z components. This and additional range information results in a raw instrument data rate of†

$$R_I \approx 800 \text{ b/s} \tag{36}$$

as in Eq. (7) for the baseline system.

Filtered Rates

A filter will be included which can provide selective adjustments in vector sampling rates by the factors

$$f_s = 1, 2, 3, 4 \text{ and } 6 \tag{37}$$

yielding effective vector sampling rates of 20, 10, 6.67, 5 and 3.3 v/s, respectively, and corresponding filter output rates of

$$R_F = 800, 400, 267, 200 \text{ and } 133 \text{ b/s} \tag{38}$$

The inclusion of this filtering can be interpreted as a form of "irreversible preprocessing" in Fig. 3 or simply modifications to the instrument output characteristics.

---

†Adjusted slightly for convenience in interpretation.

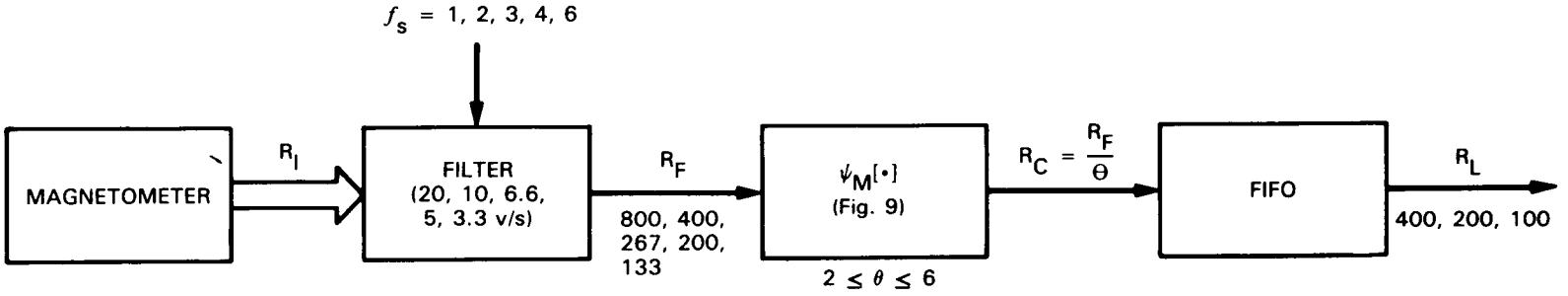

Fig. 11. Open Loop CRAF Compressor Block Diagram.

Code Operator, $\psi_M[\cdot]$

The noiseless coder from Fig. 9 (using $\psi_\ell[\cdot] \equiv$ Option C in Table 2) will be applied to the output of the sampling filter. We denote

$$R_C \quad \text{b/s} \tag{39}$$

as the output rate of the compressor and

$$\theta \tag{40}$$

as the compression factor

$$\theta = \frac{R_F}{R_C} \tag{41}$$

But θ is also

$$\theta = \frac{\text{Uncompressed b/v}}{\text{Compressed b/v}} \tag{42}$$

which we know will vary over time as data characteristics vary. But for the same segment of data, a change in sampling rate will not significantly alter the per vector performance of $\psi_M[\cdot]$. Thus θ will remain roughly the same for each sampling rate (when applied to the same segment of data). Rewriting Eq. (41) we see that

$$R_C = \frac{R_F}{\theta} \tag{43}$$

will vary proportionately with changes in filter output rate R_F.

For example, if $\psi_M[\cdot]$ generates $R_C = 300$ b/s when $R_F = 800$ b/s (20 v/s), it will generate $R_C \approx 150$ b/s if the filter rate is reduced by a factor of two to $R_F = 400$ b/s (10 v/s).

Link Rates

The communication link rate

$$R_L \quad \text{b/s} \tag{44}$$

which is made available to the magnetometer will **not be fixed**. In addition to the primary rate of 400 b/s, the magnetometer will be forced to operate with as little as 200 and 100 b/s during various phases of the mission.

Buffer

A "first-in first-out" (FIFO) buffer will be included to handle:

1) The variability in noiseless coder output rate, R_C; and

2) The variability and limitations imposed by changes in link rates, R_L.

The FIFO buffer accumulates data from the noiseless coder. Simultaneously, it also continually transfers the **oldest** of this accumulated data to the data link at a rate of R_L b/s.

If the average compressed rate, R_C, exceeds the available link rate, R_L, the buffer will gradually fill up. If it did fill up, subsequent data from $\psi_M[\cdot]$ would be lost until enough room was provided by transfers of old data. The complete loss of data in this manner is viewed as **quite unacceptable.**

But note that if $R_C < R_L$ the buffer fullness will decrease. Thus, we should be able to avoid buffer overflow by adjusting the filter sampling rate downward to lower R_F and hence $R_C = R_F/\theta$ as needed. The latter approach is discussed next.

CLOSED LOOP COMPRESSION SYSTEM

The block diagram of Fig. 11 has been modified to include a feedback loop to the filter, controlled by a measure of "buffer fullness." The result is shown below in Fig. 12.

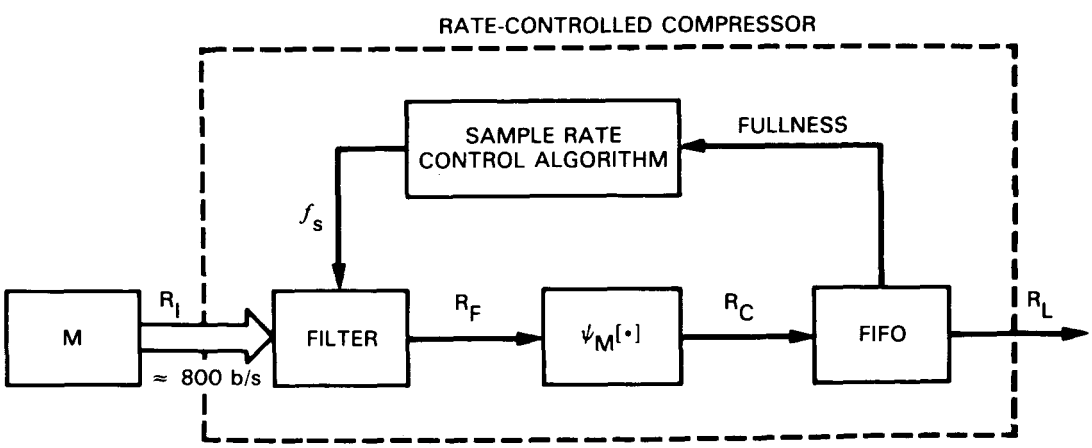

Fig. 12. Rate-Controlled Compressor.

Although a precise algorithm has not yet been specified, the basic idea is to switch the filter to a lower sampling rate (lowering R_F) whenever the buffer might overflow. Conversely, the sampling rate can be increased (higher R_F) when the buffer fullness gets too low. The fullness level itself, as well as the rate of change in buffer fullness, can be used as control parameters.

Average Effect

The long-term statistical effect of this control strategy is to maximize the average sampling rate. Lets look at a concrete example.

Suppose the available link rate is R_L = 200 b/s where an uncompressed data stream would require the sampling rate to be reduced to only 5 v/s. If the compression factor achieved were 4:1, a constant buffer level could be maintained at the full 20 v/s while matching the link rate R_L = 200 b/s.

If the compression factor were only 2:1, a constant buffer level could be maintained only if the sampling rate were reduced to 10 v/s. At any inbetween compression factor, the feedback mechanism would automatically cause switching between sampling rates **such that** the overall average compressed rate R_C (which is also the buffer input rate) equals the available link rate R_L, 200 b/s (which is the buffer output rate).

At a 3:1 compression factor, about half of the transmitted data would be at 20 v/s and half would be at 10 v/s.

Tests of this form were performed using 1985 observation data of the turbulent tail regions of Comet Giacobini-Zinner[4]. Similar average results with stable buffer levels were observed.

Transient Advantages

As these tests indicated, it is rare that the compression factor achieved actually dips below 2:1. At a link rate of 400 b/s, a constant buffer level can be maintained at the full 20 v/s sampling rate.

A number of interesting magnetometer events can produce brief bursts of very active data with corresponding compression factors dipping below 2:1[4]. But if the buffer **maintenance level** under normal conditions ($\theta \geq 2$) is kept fairly low, the remainder of the buffer can be used to **absorb** a good part of such a burst at the full 20 v/s sampling rate.

More complex strategies would even allow portions of such bursts to be preserved at high sampling rates at the lower link data rates.

OTHER MODES

The discussion above describes the principal CRAF magnetometer application of noiseless coding. Other modes of operation will be discussed in later reports.

V. BREADBOARD

A breadboard coder/decoder of $\psi_M[\cdot]$ in Fig. 9 (using $\psi_\ell[\cdot]$, Option C in Table 2) was built and tested by Bruce Parham using an Intel 8086 microprocessor[11]. The breadboard coder required roughly 1000 bytes of memory for instructions and another 1000 bytes for internal buffers. A decoder was about half the size. Both handled throughput data rates of 20 kb/s, far higher than the allowable rates on CRAF or MO.

REFERENCES

[1] E. Smith, B. Connor, and G. Foster, "Measuring the Magnetic Fields of Jupiter and the Outer Solar System," **IEEE Transactions on Magnetics, Vol. MAG-11, No. 4**, pp. 962-980, July 1975.

[2] A. Frandsen, et. al., "The ISEE-C Vector Helium Magnetometer," **IEEE Transactions on Geoscience Electronics, Vol. GE-16, No. 3**, pp. 195-198, July 1978.

[3] A. Balogh, et. al., "The Magnetic Field Investigation on ISPM," **ESA SP 1050**, ESTEC, Noordwiyk, Netherlands, July 1983.

[4] B. T. Tsurutani, et. al., "Magnetic Field Investigation for the Comet Rendezvous Asteroid Flyby Mission," **Proposal to the National Aeronautics and Space Administration**, Jet Propulsion Laboratory, Pasadena, California, November 11, 1986.

[5] R. F. Rice and J. Lee, "Noiseless Coding for the Gamma Ray Spectrometer," **JPL Publication 85-53**, Jet Propulsion Laboratory, California, June 1, 1985.

[6] R. F. Rice, "Some Practical Universal Noiseless Coding Techniques," **JPL Publication 79-22**, Jet Propulsion Laboratory, Pasadena, California, March 15, 1979.

[7] R. F. Rice and J. Lee, "Some Practical Universal Noiseless Coding Techniques, Part II," **JPL Publication 83-17**, Jet Propulsion Laboratory, Pasadena, California, March 1, 1983.

[8] R. F. Rice, et. al., "Block Adaptive Rate Controlled Image Data Compression," **Proceedings of 1979 National Telecommunications Conference**, Washington, D.C., November 1979.

[9] R. F. Rice and A. P. Schlutsmeyer, "Data compression for NOAA weather satellite systems," **Proceedings, 1980 SPIE Symposium, Vol. 249**, San Diego, California, July 1980.

[10] Chris Russell, private communication.

[11] O. Parham, private communication.

TECHNICAL REPORT STANDARD TITLE PAGE

1. Report No. JPL Pub 87-19	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle Noiseless Coding for the Magnetometer		5. Report Date June 15, 1987 6. Performing Organization Code
7. Author(s) Robert F. Rice Jun-Ji Lee		8. Performing Organization Report No. JPL Pub 87-19
9. Performing Organization Name and Address JET PROPULSION LABORATORY California Institute of Technology 4800 Oak Grove Drive Pasadena, California 91109		10. Work Unit No. 11. Contract or Grant No. NAS7-918 13. Type of Report and Period Covered JPL Publication
12. Sponsoring Agency Name and Address NATIONAL AERONAUTICS AND SPACE ADMINISTRATION Washington, D.C. 20546		14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

Future unmanned space missions will continue to seek a full understanding of magnetic fields throughout the solar system. Severely constrained data rates during certain portions of these missions could limit the possible science return. This publication investigates the application of universal noiseless coding techniques to more efficiently represent magnetometer data without any loss in data integrity. Performance results indicated that compression factors of 2:1 to 6:1 can be expected. Feasibility for general deep space application was demonstrated by implementing a microprocessor breadboard coder/decoder using the Intel 8086 processor.

The Comet Rendezvous Asteroid Flyby mission will incorporate these techniques in a buffer feedback, rate-controlled configuration. The characteristics of this system are discussed.

17. Key Words (Selected by Author(s))	18. Distribution Statement
Spacecraft Instrumentation Communications Computer Programming and Software Information Theory	Unlimited/unclassified

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified		