A BIT-SAVING ENCODING SCHEME FOR A SET OF MONOTONIC NUMBERS

The question to which this communication presents a solution is the following: If it is desired to transmit a set of numbers which have the property that each number is no smaller (or alternately larger) than its predecessor, is there any way to take advantage of this property to transmit the numbers with fewer bits?

Suppose we wish to transmit m n-bit numbers which are known to be monotonically increasing. We shall use a process akin to bit-plane encoding [1]. Arrange the numbers in an array as illustrated in Figure 1. Let k be the least integer such that $m < 2^k$. Since the numbers are monotonically increasing, we know that the first row (i.e. the row containing the most significant bit of each number) can have at most one transition from a string of binary zeros to a string of binary ones. With a k - bit number we can specify the location of this single change. (If the first row contains no binary zeros, the transmitted number would be 0, and, if all zeros, it would be m.) If the number of data words, m, is more than two this method of encoding will always save bits since the number of bits needed in the encoded system, k, is always smaller than the number of bits required for direct transmission, m. The bigger m is the greater the saving of bits.

The second row can contain at most three transitions between strings of binary zeros and strings of binary ones. but if all

To illustrate this encoding process, let us first consider a set of 255 5-bit numbers: m=255, n=5, and k=8. Then the number of bits required to encode the last row is $2^{n-1}k=(2^{5-1})\cdot 8=128$. Since this number is less than the number of bits required to transmit the row directly we can encode all 5 rows. Substituting in (1) we see that this takes $(2^5-1)8=248$ bits. The direct transmission process takes 255x5 = 1275 bits, resulting in a saving of 1027 bits (a reduction of better than 80%).

As an illustration of a case where not all rows would be encoded, let us consider a set of 15 5-bit numbers: m=15, n=5, and k=4. It takes 2k=8 bits to encode the second row and 4k=16 bits to encode the third row thus only the first two rows would be encoded: j=2. Substituting in (2) we see that this takes $(2^2-1)\ 4+(5-2)15=57$ bits. This is a saving of 18 bits (or 24%) over the direct transmission of 75 bits.

The entire discussion is equally valid if the set of numbers were known to be monotonically decreasing by merely interchanging the role of binary ones and zeros.

A similar encoding scheme can be devised for single-peaked sets of data points (such as the temperature during the course of a day). The only difference between this and the previous analysis is that in this case each successive row of bits can have more transitions between strings of zeros and ones than in the monotonic case, in fact the j-th row can have 2^j more changes than the j-1st row. It can be shown that the number of bits needed to encode a set of m n-bit numbers which increase monotonically three changes were present the second one would occur at the same position as the change of the first row. Thus only 2k additional bits are needed to specify the locations of these two significant changes. This encoding "pays off" if $2k < m$. Continuing this process, we find that the j-th row has at most 2^j-1 changes $2^{j-1}-1$ of which coincide with the changes of the (j-1)-st row. Thus, there are $[(2^j-1)-(2^{j-1}-1)] = 2^{j-1}$ new changes which we can specify by sending $2^{j-1}k$ bits. This encoding process again "pays off" if $2^{j-1}k < m$. Thus, if $2^{n-1}k < m$ the encoding process "pays off" on all n rows. We can then transmit the mn-bits worth of information using only:

$$\sum_{j=1}^{n} 2^{j-1}k = (2^n-1)k \quad \text{bits} \tag{1}$$

If, however, $2^{j-1}k < m$ but $2^j k \geq m$, all rows from the (j+1)-st on down to the last would be more efficiently transmitted directly. In this case we can transmit the mn-bits worth of information using only:

$$\sum_{i=1}^{j} 2^{i-1}k + (n-j)m$$

$$= (2^j-1)k + (n-j)m \text{ bits} \tag{2}$$

This encoding process always saves some bits for three or more data words. This is true since there is never a loss in encoding any row and there is a gain in encoding the first row.

to a maximum and then decrease monotonically is:

$$(2^{j+1}-1)k + (n-j)\ m \text{ bits} \tag{3}$$

Where, as before, k is the number of bits required to specify the change between a string of binary zeros and a string of binary ones and j is the number of rows that can be encoded with a saving of bits. For nine or more data words this scheme will save bits.

The percentage of bits saved by this encoding scheme over direct transmission is shown in Figure 2 as a function of the number of sample points (or data words) for both types of data assuming 7-bit words.

It should be noted that this method of encoding does not destroy the information; all of the original data can be reconstructed exactly.

James W. Snively, Jr.
Computer Section
Flight Data Systems Branch
Goddard Space Flight Center
Greenbelt, Maryland

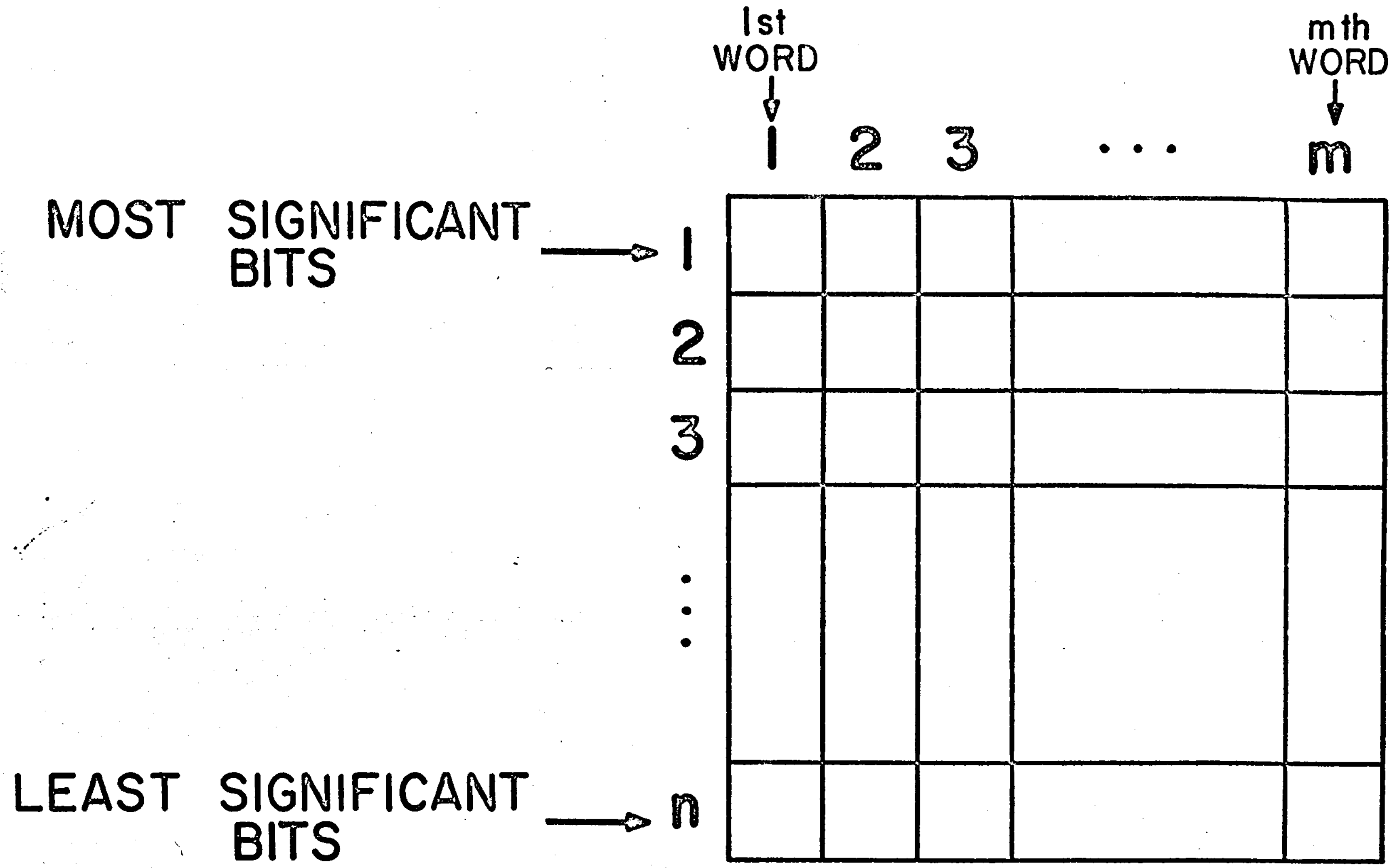
Reference

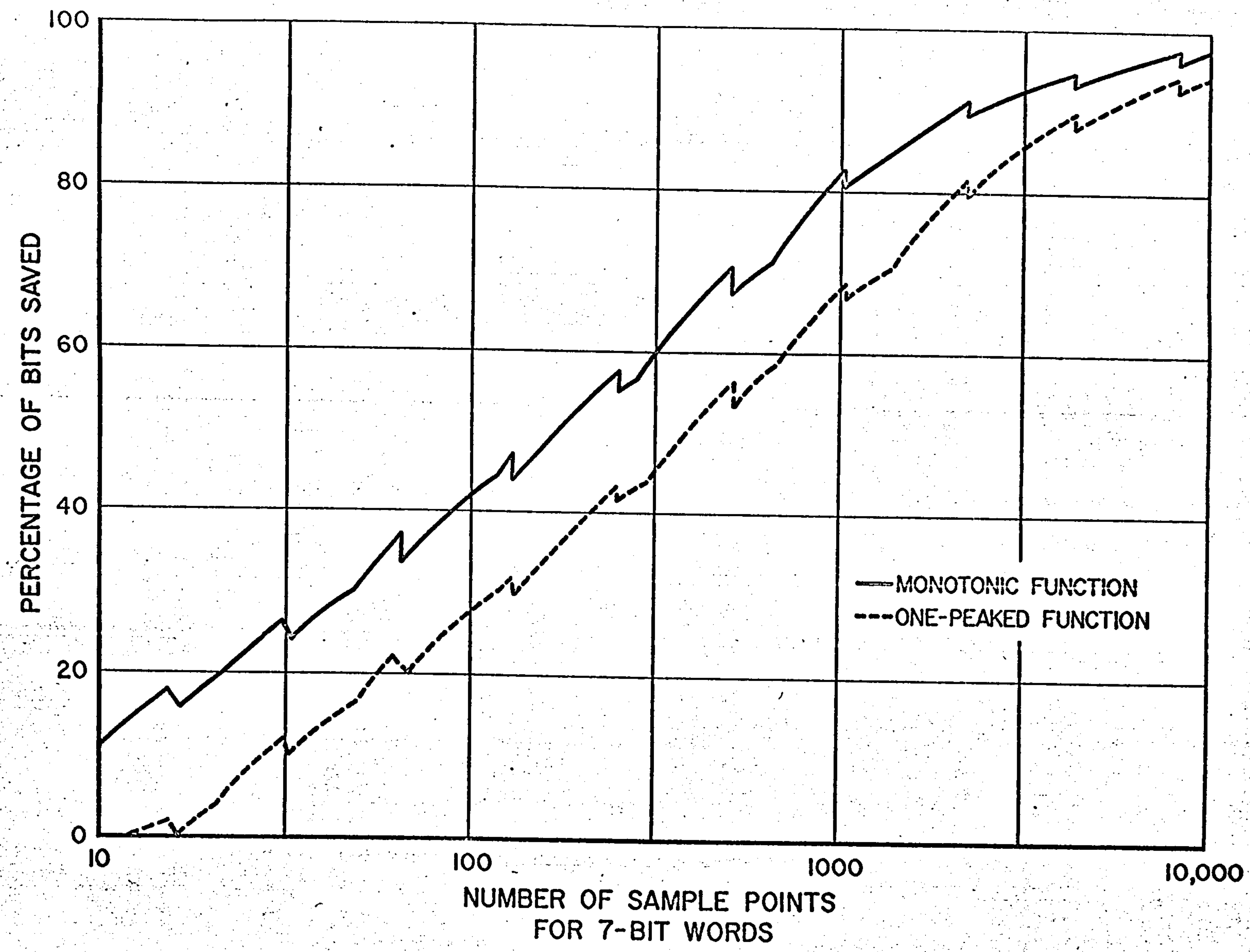
[1] Jay W. Schwartz, "Data Processing in Scientific Space Probes" Ph.D. Thesis Yale University, Dept. of Engineering and Applied Science, Sept. 1963.

Figure Captions

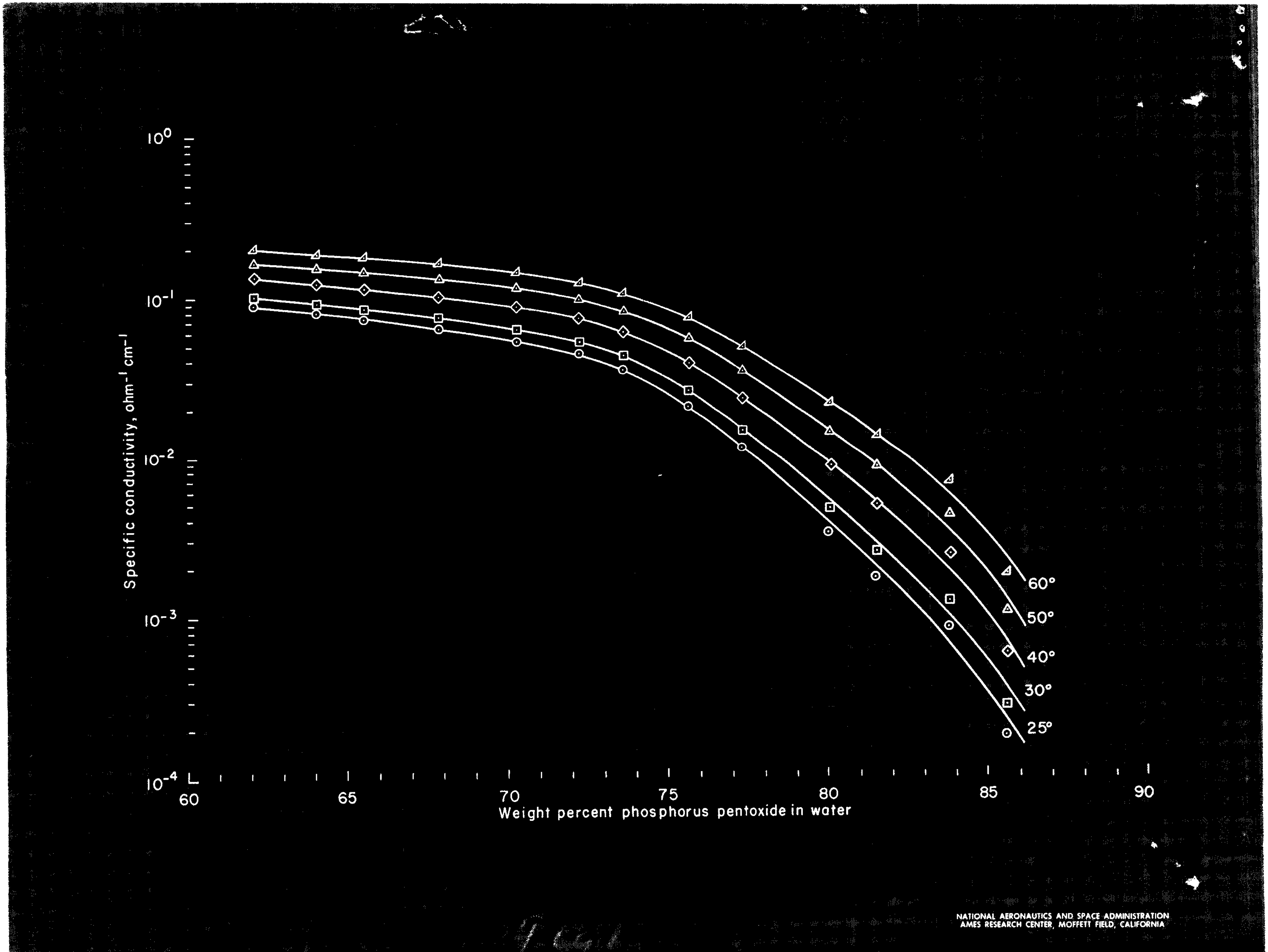
Figure 1 - Arrangement of data words for bit plane encoding.

Figure 2 - Percentage of bits saved versus number of sample points for monotonic and single-peaked data sets.


1st WORD
mth WORD
1
2
3
...
m
MOST SIGNIFICANT BITS
LEAST SIGNIFICANT BITS
1
2
3
...
n



PERCENTAGE OF BITS SAVED
100
80
60
40
20
0
10
100
1000
10,000
NUMBER OF SAMPLE POINTS FOR 7-BIT WORDS
MONOTONIC FUNCTION
ONE-PEAKED FUNCTION



Specific conductivity, ohm⁻¹ cm⁻¹
10⁰
10⁻¹
10⁻²
10⁻³
10⁻⁴
60
65
70
75
80
85
90
Weight percent phosphorus pentoxide in water
60°
50°
40°
30°
25°
NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
AMES RESEARCH CENTER, MOFFETT FIELD, CALIFORNIA