DSN Performance Tests of a Maximum Likelihood Decoder

J. M. Urech
Station Director, Cebreros, Spain

L. D. Vit
Operations Manager for the Robledo Station

C. A. Greenhall
DSN Systems Engineering Office

Viterbi decoding tests were carried out at DSS 62, Madrid, Spain. Results of bit error rate, burst statistics, and estimation of signal-to-noise ratio are presented.

I. Introduction

The present report[1] covers the results of the Madrid DSN engineering task: DSN Performance for Convolutional Decoding. This work was undertaken by personnel from Deep Space Stations (DSSs) 62 and 63 at Madrid, Spain, in a joint effort with Section 430. A preliminary report has already appeared (Ref. 1).

The objective of the task was to determine the performance of the DSN in convolutional coding when using a maximum likelihood decoder.

The study required the integration of a maximum likelihood convolutional encoder-decoder model Linkabit LV7015 into the DSS Telemetry and Command Data Handling Subsystem (TCD) and the evaluation of its performance at a medium data rate. It also included the development of the appropriate testing software and a real-time performance estimator algorithm.

[1]Based on an internal report by the first two authors.

The task began with the receipt of the LV7015 unit at the end of June 1975. After its physical integration into the DSS 62 TCD Subsystem, a preliminary testing and calibration phase was carried out in parallel with the test software development. The actual system testing was initiated early in October 1975 and was terminated in December 1975. The evaluation of the system performance was carried out simultaneously with the data gathering.

This report covers the overall task results including the integration phase, algorithm development, software description, and test results.

II. Test Plan

The test strategy was to determine telemetry performance as a function of several ratios of total power to noise spectral density (P_t/N_o) and different modulation indexes for each P_t/N_o (these values are related to the Mariner Jupiter-Saturn Mission). The tests were run at

2560 bits per second with an actual error rate of approximately $5 \cdot 10^{-5}$, using Block III receiver (RCVR) and Block III and subcarrier demodulator assembly (SDA).

The tests were grouped into blocks corresponding to specific values of P_t/N_o and subdivided into runs for different values of modulation indexes (Table 1).

Figure 1 shows a reordering of runs as a function of the ratio of bit energy to noise spectral density (E_b/N_o). Most of the runs are in the 5 to 6 dB range where the theoretical bit error rate ranges approximately from 10^{-6} to 10^{-8}.

The analysis of the test results is presented in Section II of this report.

A. Test Configuration

The main configuration characteristics are summarized as follows:

1. Simulation Conversion Assembly (SCA) Configuration

SCA control:	Manual
Bit Rate:	2560 bits/s
Bit pattern:	Repetitive 111010
Modulation:	Biphase
Subcarrier:	22.5 kHZ
Data Type:	Fixed
Data Format:	Uncoded NRZ level

2. RF configuration

Maser:	To cold start
S-Band channel:	20
RCVR bandwidth:	Narrow
Automatic gain control bandwidth:	Narrow
SDA bandwidth:	Per test plan
Carrier suppression:	Using 50-MHz Y-factor

The carrier suppression adjustment allowed an accuracy of about 0.1 dB. The SDA-RCVR phasing was performed at a strong signal level and by adjusting the SDA modulation index attenuator for 280 mV peak to peak.

The E_b/N_o calibrations were also done by means of the Y-factor.

3. Digital configuration. The digital configuration is summarized in Fig. 2. A detailed explanation of Phases I and II is given in Subsection II-B-3.

B. Test Software Description

A primary objective of this DSN engineering task was to evaluate DSN performance with convolutionally coded data when using the LV7015 unit. Some DSN projects already use convolutional coding but are decoded by a Fano-type algorithm. The present case, which is oriented to the Mariner Jupiter-Saturn Mission, uses convolutional data decoded by the Viterbi maximum likelihood decoding criteria. Then, since all the existing testing software was designed for a sequential Fano decoder, a new program was developed for the evaluation of the DSN performance with the Viterbi algorithm.

The main concern when decoding with the Viterbi algorithm is the decoded data bit error rate. With the Fano algorithm the main concern is the probability of erasing a block of coded data, that is, the probability of being unable to process a data frame before the next frame is ready for decoding. Therefore, in the present case the overall testing philosophy consists precisely in analyzing the bit error patterns.

1. Bit error characteristics. The Viterbi decoder algorithm does not proceed on a per block basis like the Fano algorithm nor does it consider past bit decisions. The decoded bits may be in error in a certain path length and yet able to remerge with a good path at a certain node and remain correct thereafter. Therefore, the decoder always proceeds forward and may depart from the correct path occasionally depending on channel noise characteristics. The bit errors occur in bursts whose characteristics are to be determined for the testing conditions. The burst approach suggests two definitions.

(1) An error-free run is a sequence of consecutive correct bits. Two different types of runs will be considered: Type 1 includes runs of length 0 to 5 ($R < 6$), and Type 2 includes runs of length 6 or greater ($R \geq 6$).

(2) An error burst is a sequence of decoded bits which begins with a bit in error, ends with a bit in error, contains only Type 1 runs, and is surrounded by Type 2 runs. The shortest burst has length 1 (a single isolated error).

The statistical analysis of runs will then distinguish between correct bits within a burst assuming there is a

run of length zero between two consecutive bits in error, and runs of correct bits not in a burst; that is, 6 or more consecutive good bits.

The main point thereafter is to identify the bits in error, proceed with their classification into bursts and runs, and then analyze the clustering of errors and correct bits within the bursts.

2. Data compression. In order to handle the previously mentioned conditions, the following approach is taken: A bit error pattern is obtained by direct comparison of the original data and the decoded bits. Therefore the *ones* in this pattern represent bits in error in the decoded data. Instead of operating directly with this binary error pattern, a preprocessing step is first carried out. The number of consecutive *ones* and consecutive *zeros* in the error pattern are counted. The binary pattern is compressed into a sequence of integers where the odd terms correspond to consecutive *zeros* in the error pattern while the terms of even order correspond to consecutive *ones* in the error pattern. Note that the alternative choice could have been made as well. However, since it is much more likely that the actual sequence will begin with a run, the former approach is selected. This preprocessing (Phase I) greatly reduces the data storage required, the search time, and also simplifies the statistical evaluation (done in Phase II). Therefore, given an error pattern of the form

$$\underbrace{0\ldots.0}_{K1}\quad\underbrace{1\ldots.1}_{K2}\quad\underbrace{0\ldots.0}_{K3}\quad\underbrace{1\ldots.1}_{K4}$$

the corresponding sequence of integers would be

$$K1, K2, K3, K4 \qquad Ki \in N$$

where N is the set of non-negative numbers.

In general, the runs of ones and the runs of zeros are transformed into a sequence of integers

$$(a_i); \qquad i \in N$$

where the subsequence of odd terms

$$(a_i); \qquad i = 2K - 1, K = 1,2,3, \cdots$$

represents the respective lengths of the runs of zeros, and the subsequence of even terms

$$(a_i); \qquad i = 2K \qquad K = 1,2,3, \cdots$$

represents the respective lengths of the runs of ones. All the considerations concerning error bursts and runs are taken directly from the sequence (a_i).

3. Software characteristics. As previously stated, the overall process is carried out in two phases.

During Phase I the data are gathered in real time as per the configuration depicted in Fig. 2. The high-speed data (HSD) blocks are classified and the decoded data are synchronized and compressed as explained in Subsection II-B-2. A quick-look display of signal level, Symbol Synchronizer Assembly (SSA) signal-to-noise ratio (SNR), normalization rate, and bit error rate (BER) is optional through TTY or line printer for a preliminary test verification. The final product of Phase I is then a classified and compressed error pattern.

The actual data analysis is performed during Phase II, and the mag tape Digital Original Data Record (DODR) recorded during Phase I is processed as follows:

(1) The mag-tape records are classified into in-sync or out-of-sync status, matrix records, statistics records and end of runs.

(2) The statistical analysis and outputs are carried out and displayed on the line printer.

III. Test Results

A brief review of the test plan setup conditions reflects a concentration of SNRs especially in the 5- to 6-dB range (Fig. 1) which corresponds to very small variations of modulation indexes for relatively close values of P_t/N_o. The resulting variations from one test to another are in most cases smaller than the intrinsic uncertainties of the evaluating algorithms. It is then difficult to determine whether the deviations relative to the expected values are due to set-up errors or actual degradations. For a better presentation of the results, a table has been assigned to each block of runs. These tables (Table 1) contain the following data extracted from the test printouts:

(1) E_b/N_o (TH): Theoretical values from the test plan.

(2) $\Delta E_b/N_o$ $(\overline{N}_c)$: The difference E_b/N_o (TH) $-$ E_b/N_o $(\overline{N}_c)$, where E_b/N_o $(\overline{N}_c)$ is the estimation of the bit energy-to-noise density through the normalization rate algorithm developed in Section IV.

(3) $\Delta E_b/N_o$ (SSA): The difference E_b/N_o (TH) $-$ E_b/N_o (SSA), where E_b/N_o (SAA) is the SNR estimator by the SSA, unbiased for values over 5 dB and incre-

mented by 3 dB to represent bit energy-to-noise density ratio.

(4) The total number of bits in the run.

(5) The bit error rate directly obtained from the error pattern.

(6) $\Delta E_b/N_o$ (BER): The difference between E_b/N_o (TH) and the value of E_b/N_o which theoretically gives the observed BER.

(7) Average length of runs ≥ 6.

(8) Average length of burst.

(9) Average number of errors per burst.

(10) Average density of errors in a burst. This is simply the average number of errors per burst/average length of burst.

The test results are analyzed next from two different aspects:

(1) In terms of system degradation.

(2) In terms of statistics on runs and bursts, and other parameters.

A. System Degradation

The system performance in terms of degradation is evaluated in three different steps:

(1) The SSA SNR which shows the performance at the decoder input.

(2) The bit error rate which is actually the basic parameter reflecting decoder performance.

(3) The normalization rate which reflects the specific characteristic of the decoder. This study is made in Section IV.

1. SSA SNR algorithm. Figure 3 represents the differences between E_b/N_o (TH) and E_b/N_o (SSA) for each run. These results are typical representations of the SSA SNR statistical variations. It may be noted for instance, that for blocks H, I and J having practically the same P_t/N_o, a slight average degradation increase is observed as the SDA bandwidth is increased. Also, there seems to be a very small increase in degradation for the highest modulation indexes in each block.

As will be emphasized later on, the SSA SNR values are very poorly correlated with the actual decoder behavior in terms of bit errors since large increments in the number of errors are not reflected by the SSA algorithm.

2. System performance versus BER. All the results concerning the bit error rate are of great importance for the system performance evaluation. The decoder behavior will be reflected in the number of errors for each test condition. Figure 4 shows the difference between E_b/N_o (TH) and E_b/N_o derived from experimental values of the bit error rate. There is a striking difference with Fig. 3 in that the BER shows a rapid increase for runs with higher modulation indexes in a given test block. This degradation effect (actually at the highest data SNR) is justified by the corresponding receiver jitter increment at lower receiver margin values. For a given total power, at higher modulation indexes the increased carrier suppression causes a slowly varying (compared to bit rate) jitter which affects system performance by introducing additional degradation. It must be noted though that the optimum working point is not to be derived from the results plotted in Fig. 4. Instead it is more convenient to analyze the behavior of the bit errors as a function of the modulation indexes and establish the optimum point by choosing (for each P_t/N_o) the carrier suppression yielding the lowest error rate. This is done in Fig. 5 for each of the test blocks.

It is clearly seen that there exists a minimum degradation point on the order of 70 degrees, while above these values the degradation due to jitter greatly overcomes the increase of data power obtained from higher carrier suppressions. Of interest is the comparison of these results with those presented on page 173 of Ref. 2; the experimental results are compatible with the theoretical model. There also appears to exist a correlation between the results in Figs. 4 and 5 as expected. The family of curves of Fig. 5 are also compatible with the test conditions in terms of total power-to-noise density ratios. Several conclusions are then available at this point:

(1) The increasing values of modulation indexes cause an increasing degradation due to jitter when surpassing a certain optimum modulation point in the range of 70 degrees.

(2) The corresponding bit error rate increase is not reflected by the SSA SNR algorithm which shows very small increments in degradation.

It must be brought out again that the close proximity of test setup conditions and the normal algorithm dispersions make a neat evaluation of the optimum conditions difficult, but approximate values are possible with an

acceptable definition. Evidently, the optimum modulation indexes could be pinned down more precisely if more tests were run between 65 and 70 degrees.

B. Statistical Results on Bursts and Runs

The results developed in the previous paragraph would be conclusive as far as optimum working conditions if the bit errors at the decoder output were randomly distributed. However, this is not the case for a maximum likelihood decoder where the errors tend to concentrate into bursts. Therefore, although an optimum point has been derived from experimental results, this situation must now be confirmed in terms of the statistical behavior of bursts and runs.

1. Theoretical considerations. The definition of burst is tied closely to the choice of code and method of decoding. The convolutional code has constraint length 7; the state of the decoder consists of the last 6 bits shifted into it. (See Ref. 3 for an exposition of Viterbi decoding.) We can consider the correct path through the trellis as being the all 0's path, and assume that all paths ultimately fail to survive except one. The state of the ultimate surviving path differs from the state of the correct path whenever there is a 1 in the state bits of the shift register, and returns to the correct state as soon as a run of 6 0's has appeared. Thus, there is a one-to-one correspondence between the sequence of bursts and the sequence of excursions of the surviving path from the correct path. An excursion M branches long will yield a burst of length $M - 6$. If there are n wrong bits in this burst, there are $M - 6 - n$ correct bits grouped in runs <6.

The fixed path memory of the decoder can cause additional errors not included in this scheme; the decoder will occasionally choose a bit not belonging to the ultimate survivor path.

2. Error bits in bursts and burst lengths versus BER. We shall now select the BER as a variable and study its effects on the number of errors in a burst and the burst length. Let

P_1 = probability that a burst has length 1

L = average length of a burst

D = average density of errors in a burst

D_i = average density of errors in the interior of a burst

= total number of interior errors/total number of interior bits

The interior of a burst consists of the burst minus its endpoints (which are always bad bits). Bursts of length 1 or 2 have empty interior. The relation

$$D = D_i\left(1 - \frac{2 - p_1}{L}\right) + \frac{2 - p_1}{L}$$

is a direct consequence of the definitions.

Figures 6, 7, and 8 are scatter diagrams of L, D, and D_i, respectively, versus BER. The L and D plots show slow increasing and decreasing trends, but D_i fluctuates about a constant level of about 0.48 for $5 \cdot 10^{-6} < \text{BER} < 2 \cdot 10^{-3}$. (For $\text{BER} < 5 \cdot 10^{-6}$ the data are scanty and the statistics poor.) More information on the behavior of the interior bits could be obtained from the distributions of runs of length <6. We have not yet done this; nevertheless, we will tentatively model a burst as having a length whose mean depends slowly on the BER, and an interior error density whose mean is constant and slightly less than ½.

3. Statistical results on runs. The average run length ($R \geq 6$)is directly related to the BER. Let $\overline{R}$ be the average length of the runs $R \geq 6$. Then $\overline{R}$ = (number of bits − total burst length)/number of $R \geq 6$. Since total burst length $<<$ number of bits, and number of $R \geq 6$ = number of bursts + 1, we get

$$\overline{R} = \frac{\text{L} \cdot \text{D}}{BER \cdot (1 + 1/N_b)} \tag{1}$$

where N_b = number of bursts. This applies to average values, but does not reflect the statistical behavior of runs. The statistical analysis carried out by the test software includes the distribution function of $R \geq 6$. These curves for Block H tests are shown in Fig. 9.

The plots seem to add some additional information to the curves in Fig. 5. The optimum range in the family of curves is not as broken as in the case of Fig. 5 and even defines a broader range of appropriate values.

The same degradation effects observed at different modulation indexes upon the BER are also affecting the average run length in a similar manner. The effect of the increasing total power-to-noise density according to the test plan is reflected as expected by longer runs of error-free bits.

However, for each family of curves the optimum choice seems, as stated before, less broken than in Fig. 5. As a

matter of fact, tests J, H, G and K yield the longest runs at modulation indexes in the range corresponding to about 71 or 72 degrees. The distribution plot of Test I appears to be quite correlated with the results in Fig. 5 and the best curve in the latter is the deepest point in the former plotting. Tests L did not yield sufficient statistics as to allow a plotting of a curve.

These run length distributions are the distributions of waiting time between bursts. As a trial hypothesis, we might suppose that the positions of the bursts form approximately a Poisson process on the time axis. If this is so, then we should have

$$P = e^{-(r-6)/(\bar{R}-6)}$$

where P is the probability that a run of length ≥ 6 is not less than r, and $\bar{R}$ is the average length of runs ≥ 6. (We pretend that we are dealing with a continuous distribution.) Treating the 6's as negligible, we rewrite the above as

$$\log_{10} \ln (1/P) = \log_{10} r - \log_{10} \bar{R} \qquad (2)$$

Figure 10 is a log-log plot of ln $(1/P)$ versus r for most of the curves of Fig. 9. The dashed lines are from Eq. (2) using the values of $\bar{R}$ from Table 1. The fit is good; we conclude that at least for $r > 10^3$ the distributions are exponential and are thus determined by their means $\bar{R}$.

It is then appropriate to plot $\bar{R}$ versus modulation index for each block. This we have done in Fig. 11. The result is essentially an upside-down version of Fig. 5, with slight differences. (Eq. (1) would lead us to suspect this.) It is apparent that either run length distributions or BER can be used to locate optimum modulation indexes.

4. Statistical results on bursts. Figure 12 represents selected burst length distribution curves[2] for Block H tests. As expected, these curves are similar and do not present significant deviations. They depend slightly on the signal-to-noise condition with longer bursts at higher bit error rates.

It must be noted, however, that the maximum burst lengths (not shown in this article) are in some cases much longer than expected. In these test runs there is a large gap between the maximum and next to maximum values. This has not been explained fully but there are a few indications that they might be caused by random malfunctions in the SSA coupler.

[2]Only the bursts of length ≥ 2 are included.

IV. System Performance Evaluator

It is convenient to derive the system performance evaluation from some parameter directly related to the decoder operation and the only parameter which may be made available to the operational program is the normalization counter. The following presentation justifies the use of the normalization values as performance evaluator.

A. Normalization Rate

The Viterbi decoder algorithm behaves basically as a progression along the trellis by pairwise comparisons of paths and the elimination of less probable paths, following a metric criterion. The pairwise comparisons are made at each bit time and the metric values are derived from the channel symbol quantizations provided by the SSA, and the branch symbols hypothesized by the so called "normalization rate" mechanism.

The normalization mechanism may be visualized as being composed of a set of 4-stage buffers which, at each bit time, are incremented by a metric value and then compared pairwise as per the trellis structure. To simplify the scheme it may be assumed that the path holding the lowest metric is considered to be the "best" path. However, during the decoding process all paths including the "best path" will accumulate metric values so as to saturate their corresponding buffers (assuming a significant noise level). In the decoding range of operation the "wrong" paths will nevertheless accumulate at a much faster rate than the "best" path. A *normalization* occurs when the logic detects that all the "wrong" paths have a high metric ($M >> 4$) and that the best path has just reached or surpassed a threshold of 4. At this time all the buffers are reduced (normalized), and this fact (normalization) is registered in a counter.

B. Theoretical Model

Figure 13 represents schematically the density function of the SSA output, together with the schemes for quantizing the output and computing the metric increment m. It is assumed that a "0" symbol was sent and that the best path also outputs a "0" at this point. If the BER is low, we can assume that for most of the time, the best path symbol agrees with the symbol that was sent. Then the accumulation of metric by the best path, i.e., the normal-

ization rate, will reflect the ratio of symbol energy to noise spectral density (E_s/N_o) seen by the decoder.

The normalization per bit N_b is just half the accumulation of metric per symbol time by the best path. In terms of the probabilities p_j shown in Fig. 13, the *mean* normalization per bit, $\overline{N}_b$, is given by

$$\overline{N}_b = 1/2 \sum_{j=1}^{4} j p_j \tag{3}$$

Let $P_e(E_s/N_o)$ be the probability of a hard-decision symbol error for any given E_s/N_o. Let $K = M/Q$, where M is the integration mean and Q the quantization interval. Then Eq. (3) becomes

$$\begin{aligned}\overline{N}_b &= 1/2 \sum_{i=0}^{\infty} P_e((E_s/N_o)(1 + i/K)^2) \\ &= f(E_s/N_o, K)\end{aligned} \tag{4}$$

Now the question arises: Will the model depart from experimental results at low E_s/N_o when the bit error rate is significant? How will the model be adapted to the system operating conditions?

In order to answer the above questions a series of tests were run with a standard configuration, sampling the normalization counter for different E_s/N_o and assuming a value of K corresponding to the optimum SDA/SSA setting of 280 mV. We have determined experimentally that this value of K is 2.5.

C. An Algorithm for Performance Evaluation

The mathematical model developed in the previous section seems to be accurate enough to constitute the basis for a performance evaluator. The requirements may be summarized as a need to evaluate decoder performance (or the system performance) from the normalization rate. A convenient performance estimator could be one which would evaluate the bit error rate or, more simply, the energy per bit-to-spectral noise density. In our case the problem will be reduced to relating the normalization counter values to the bit energy-to-spectral noise density, E_b/N_o.

The function f in Eq. (4) was evaluated numerically for a range of E_s/N_o and for several values of K. Then, with K set equal to the optimum value, the function was inverted graphically to give $E_b/N_o = E_s/N_o + 3$ dB as a function of $\overline{N}_b$. However, since the normalization counter is transferred to the operational program every 192 bits, it was thought that it would be preferable to use normalization counts ($\overline{N}_c$) instead of the normalization rate ($\overline{N}_b$) as the variable. Thus, a final change was made where

$$\overline{N}_c = 192 \times \overline{N}_b$$

and finally

$$E_b/N_o, dB = g(\overline{N}_c)$$

was obtained.

For the practical purposes of using the relationship as a computerized algorithm, a rational function was fitted to the numerical values of $g(\ )$. The final expression adopted for the algorithm was then

$$E_b/N_o, \text{dB} = \frac{2.9664}{\overline{N}_c + 0.08} + 5.1218 - 0.2252\,\overline{N}_c \tag{5}$$

This expression will, therefore, convert the normalization counts as transferred from the decoder into the corresponding channel E_b/N_o, dB. Figure 14 is a plot of expression (5) and is compared to the values of $g(\ )$. The fit has an error lower than 0.3 dB in the range $1 < \overline{N}_c < 15$. Although the fit is poor for $\overline{N}_c < 1$, the approach is not useful anyway in this region. The statistics become very poor at E_b/N_o over 7 dB, since extremely few normalizations will occur and the conversion into E_b/N_o becomes less relevant.

D. Experimental Results

The experimental results obtained from the test runs are now analyzed to prove the practical validity of the algorithm. The E_b/N_o values derived from the normalization rate are compared to the corresponding SSA SNR algorithm. Fig. 15 shows the difference in dBs between the normalization algorithm values and the SSA values. The normalization algorithm values have been adjusted to correct the curve fitting error of Fig. 14. For most of the runs, the normalization and SSA estimates agree within 0.2 dB. The conclusion is that the normalization algorithm behaves very much as the SSA algorithm.

Finally, in Fig. 16 we plot BER versus E_b/N_o ($\overline{N}c$) for the test runs having optimum modulation index (lowest BER for each block). The SNR values have again been corrected according to the curve fit error in Fig. 14. The plotted points show rather good agreement with a theoretical curve of BER versus E_b/N_o. Thus the normalization rate algorithm seems to give a good estimate of the SNR that the decoder actually sees.

Acknowledgments

This study represents the combined effort of several members of the Madrid Space Station staff including A. Chamarro (Operational Software Integration and Linkabit Model LV 7015 Integration); J. L. Morales (Testing Software Development); M. S. Cristobal and A. M. Rosich (Testing Preparation and Coordination); and J. L. Alonso, A. Cancela, and F. Sanz (Hardware Interface Design and Implementation). The contributions of the DSS 62 personnel who configured and ran the tests are also acknowledged.

References

1. Urech, J. M. et al., "Preliminary Results of DSN Performance for Convolutional Codes With a Viterbi Decoder," in *The Deep Space Network Progress Report 42-32*, pp. 222-240, Jet Propulsion Laboratory, Pasadena, Calif., April 15, 1976.

2. Edelson, R. E., ed., *Telecommunications Systems Design Techniques Handbook*, Technical Memorandum 33-571, Jet Propulsion Laboratory, Pasadena, Calif., 1972.

3. Viterbi, A., "Convolutional Codes and Their Performance in Communications Systems," *IEEE Transactions on Communications Technology*, Vol. COM-19, No. 5, 1971.

Table 1. Summary of test conditions and results (1.2E3 = $1.2 \cdot 10^3$)

Run	Mod. index, deg	E_b/N_o (Th)	$\Delta E_b/N_o$ ($\overline{N}_c$)	$\Delta E_b/N_o$ (SSA)	No. of bits	BER	$\Delta E_b/N_o$, BER	Average run length ≥ 6	Average burst length	Average no. of errors per burst	Average density of errors in burst
Block H, P_T/N_0 = 39.67 dB, RCVR/SDA bandwidths: 12 Hz/Narrow											
1	55	3.85	0.75	0.72	9.9E6	7.31E-4	0.7	5.16E3	6.06	3.77	0.623
2	65	4.73	0.87	0.92	9.0E7	1.13E-4	1.1	3.05E4	5.61	3.45	0.615
3	69	4.99	0.71	0.57	9.0E6	1.19E-5	0.6	1.96E5	3.66	2.38	0.652
4	70	5.05	0.65	0.48	4.9E7	2.04E-5	0.9	1.87E5	5.96	3.83	0.643
5	70.6	5.08	0.94	0.97	9.1E7	4.81E-5	1.1	7.67E4	5.82	3.69	0.634
6	71	5.10	0.57	0.50	4.4E7	1.35E-5	0.8	2.58E5	5.76	3.50	0.608
7	72	5.15	0.83	0.77	9.9E6	2.39E-5	1.0	1.54E5	5.80	3.74	0.644
8	73	5.20	0.85	0.80	9.4E6	8.11E-5	1.4	4.71E4	6.15	3.84	0.624
9	74	5.24	0.84	0.79	1.0E7	6.86E-5	1.4	6.19E4	6.94	4.27	0.615
10	76	5.33	0.80	0.69	9.4E6	7.13E-5	1.6	6.08E4	7.46	4.36	0.585
11	78	5.40	1.21	0.91	1.0E7	3.32E-4	2.1	1.68E4	9.50	5.59	0.588
12	80	5.45	0.88	0.75	9.0E6	7.37E-4	2.3	8.4E3	10.7	6.20	0.577
Block I, P_T/N_0 = 39.77 dB, RCVR/SDA bandwidths: 12 Hz/Medium											
1	55	3.95	0.64	0.62	1.0E7	4.13E-4	0.6	9.3E3	6.28	3.84	0.612
2	65	4.83	0.83	0.83	9.7E6	4.67E-5	0.9	7.7E4	5.84	3.62	0.621
3	69	5.09	0.70	0.56	8.8E6	8.0E-6	0.6	3.7E5	4.58	3.08	0.672
4	70	5.15	0.89	0.84	8.1E7	2.5E-5	1.0	1.4E5	5.42	3.51	0.646
5	70.7	5.19	0.82	0.80	7.1E7	2.58E-5	1.1	1.5E5	6.13	3.88	0.633
6	71	5.20	0.82	0.78	6.9E7	3.88E-5	1.2	9.7E4	6.05	3.77	0.623
7	72	5.25	0.91	0.82	1.3E7	5.17E-5	1.3	7.6E4	6.44	3.95	0.614
8	73	5.30	0.82	0.73	1.1E7	5.17E-5	1.4	8.4E4	7.18	4.38	0.609
9	74	5.34	0.81	0.75	1.0E7	4.11E-5	1.4	9.8E4	6.71	4.07	0.606
10	76	5.43	0.82	0.74	9.0E6	4.91E-5	1.5	9.8E4	7.86	4.86	0.619
11	78	5.50	1.67	0.94	7.4E6	5.36E-4	2.3	1.0E4	9.08	5.37	0.591
12	80	5.55	1.30	1.14	9.0E6	1.95E-3	2.8	3.8E3	13.0	7.41	0.572
Block J, P_T/N_0 = 39.82 dB, RCVR/SDA bandwidths: 12 Hz/Wide											
1	55	4.00	0.82	0.67	1.1E7	6.39E-4	0.8	6.0E3	6.30	3.84	0.609
2	65	4.88	0.91	0.95	1.1E7	7.09E-5	1.1	5.18E4	5.63	3.69	0.655
3	69	5.14	1.16	0.81	9.5E7	3.45E-5	1.1	9.89E4	5.43	3.42	0.629
4	70.8	5.24	1.01	1.0	4.7E7	4.56E-5	1.3	8.28E4	6.06	3.78	0.624
5	71	5.25	1.02	0.91	4.5E7	3.4E-5	1.2	1.18E5	6.41	4.02	0.627
6	72	5.30	0.91	0.81	9.4E6	4.71E-5	1.4	8.3E4	6.54	3.94	0.603
7	73	5.35	0.92	0.81	9.1E6	3.48E-5	1.3	1.02E5	5.59	3.59	0.642
8	74	5.39	0.91	0.88	9.4E6	5.43E-5	1.5	7.7E4	6.85	4.22	0.616
9	76	5.48	1.02	0.87	9.6E6	6.46E-5	1.5	6.28E4	6.63	4.08	0.616
10	78	5.55	1.28	0.93	9.0E6	1.97E-4	2.1	2.44E4	8.17	4.82	0.590
11	80	5.60	1.24	1.07	9.2E6	1.58E-3	3.0	4.53E3	12.5	7.16	0.574

Table 1 (contd)

Run	Mod. index, deg	E_b/N_o (Th)	$\Delta E_b/N_o$ ($\overline{N}_c$)	$\Delta E_b/N_o$ (SSA)	No. of bits	BER	$\Delta E_b/N_o$, BER	Average run length ≥ 6	Average burst length	Average no. of errors per burst	Average density of errors in burst
Block G, $P_T/N_0 = 40.26$ dB, RCVR/SDA bandwidths: 12 Hz/Medium											
1	55	4.44	0.61	0.62	9.9E6	7.92E-5	0.7	4.4E4	5.40	3.50	0.648
2	65	5.32	0.82	0.65	1.0E7	6.55E-6	0.8	5.5E5	6.05	3.81	0.630
3	69	5.58	0.87	0.61	8.4E6	3.55E-6	0.9	7.67E5	4.50	3.00	0.666
4	70	5.64	0.97	0.69	8.0E6	4.48E-6	1.0	4.66E5			
5	71	5.69	0.83	0.63	7.4E7	5.47E-6	1.1	6.7E5	6.23	3.70	0.594
6	72	5.74	0.87	0.59	4.3E7	6.33E-6	1.2	5.8E5	6.11	3.72	0.610
7	73	5.79	0.98	0.76	3.9E7	2.63E-5	1.7	1.8E5	7.57	4.76	0.628
8	74	5.83	0.97	0.72	1.1E7	2.02E-5	1.6	2.0E5	6.41	4.12	0.642
9	75	5.88	1.2	0.64	9.0E6	9.03E-6	1.4	4.2E5	6.14	3.98	0.648
10	78	5.99	1.29	0.80	9.1E6	8.87E-5	2.3	5.5E4	7.97	4.91	0.616
11	80	6.04	1.02	0.77	8.6E6	3.0E-4	2.6	1.9E4	9.89	5.71	0.577
Block K, $P_T/N_0 = 41.26$ dB, RCVR/SDA bandwidths: 12 Hz/Medium											
1	55	5.44	0.83	0.78	9.6E6	8.37E-6	1.0	5.8E5	8.58	5.17	0.602
2	65	6.32	1.0	0.72	8.5E6	2.29E-7	0.8	4.3E6	2.00	2.00	1.00
3	72	6.74	0.98	0.46	9.0E7	1.13E-7	1.1	2.2E7	5.60	3.30	0.589
4	73.5	6.81	1.27	0.81	8.6E7	3.88E-7	1.5	6.2E6	3.75	2.59	0.691
5	74	6.83	0.96	0.70	8.5E7	3.77E-7	1.5	6.25E6	3.78	2.54	0.674
6	75	6.88	1.22	0.94	9.3E6	6.31E-7	1.7	3.16E6	5.50	3.02	0.549
7	76	6.92	0.89	0.54	9.4E6	1.73E-6	2.0	3.1E6	11.0	8.00	0.727
8	77	6.95	0.87	0.56	9.0E6	1.54E-6	2.0	1.5E6	4.36	2.77	0.636
9	80	7.04	1.31	0.85	8.5E6	1.45E-4	2.5	3.8E4	9.44	5.53	0.587
Block L, $P_T/N_0 = 42.76$ dB, RCVR/SDA bandwidths: 12 Hz/Medium											
1	55	6.94	0.80	0.47							
2	65	7.82	0.23	0.42							
3	73	8.29	−0.52	0.39							
4	74	8.33	−4.1	−0.37							
5	75	8.38	0.33	0.72							
6	75.2	8.38	0.15	0.63							
7	76	8.42	0.21	0.69							
8	77	8.45	0.27	0.77							
9	78	8.49	1.07	0.85							
10	80	8.54	−0.03	0.69		2.3E-6	3.7				

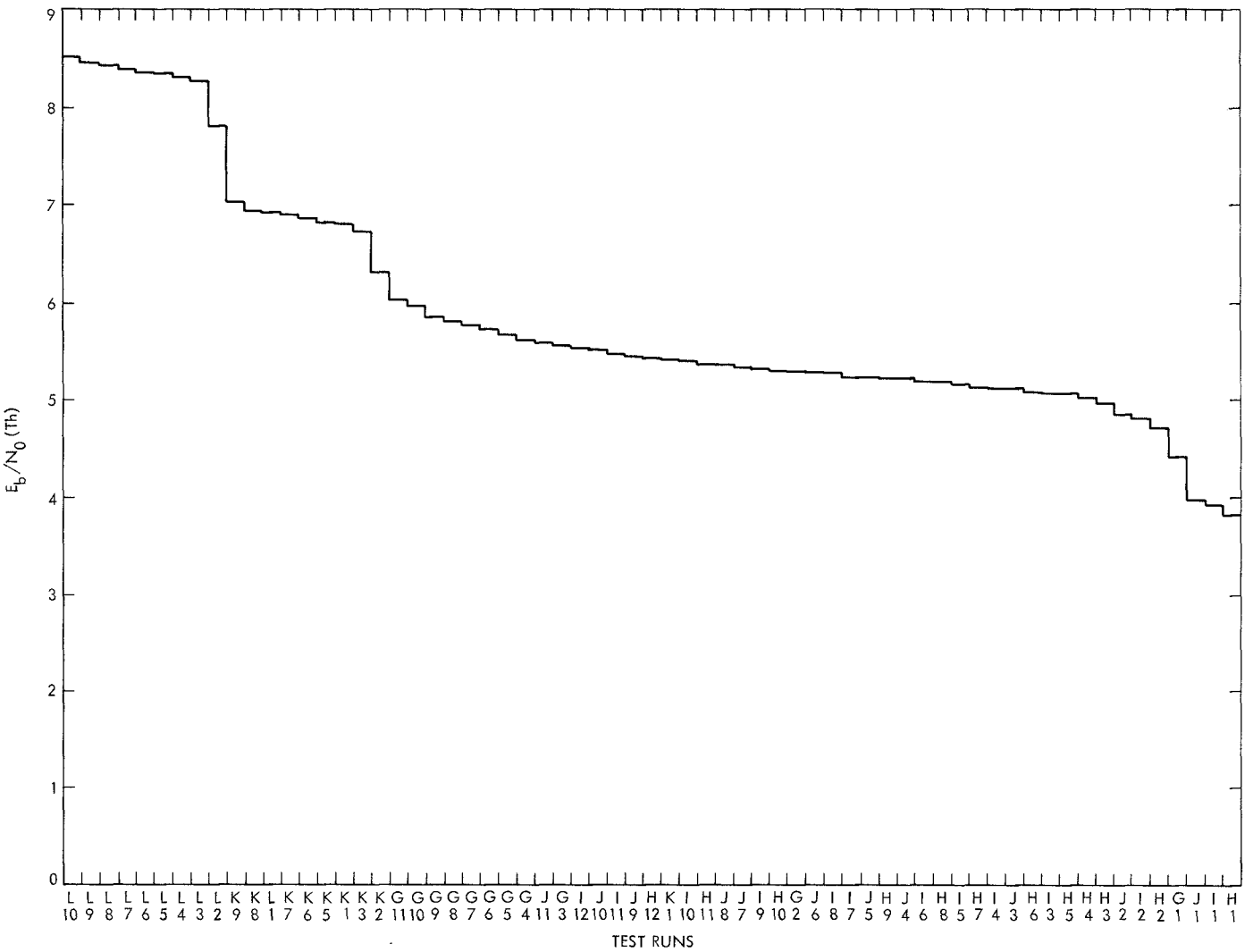

Fig. 1. Display of E_b/N_o

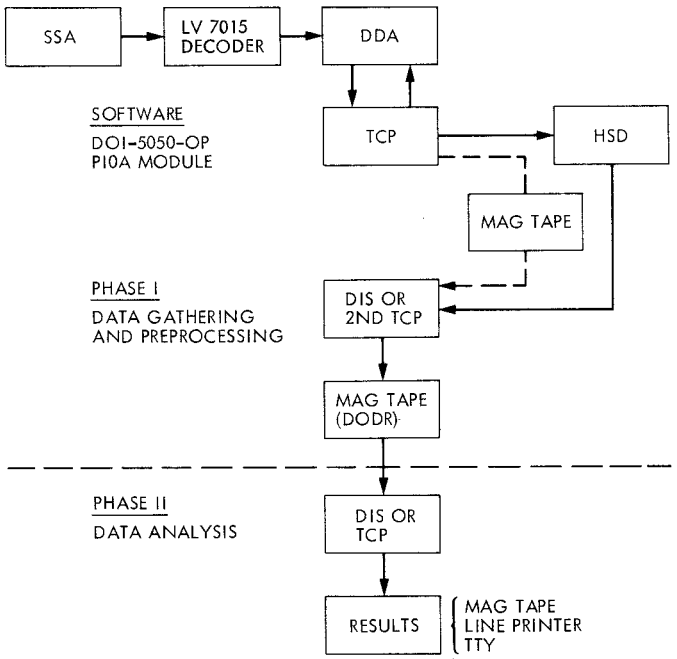

Fig. 2. Digital configuration

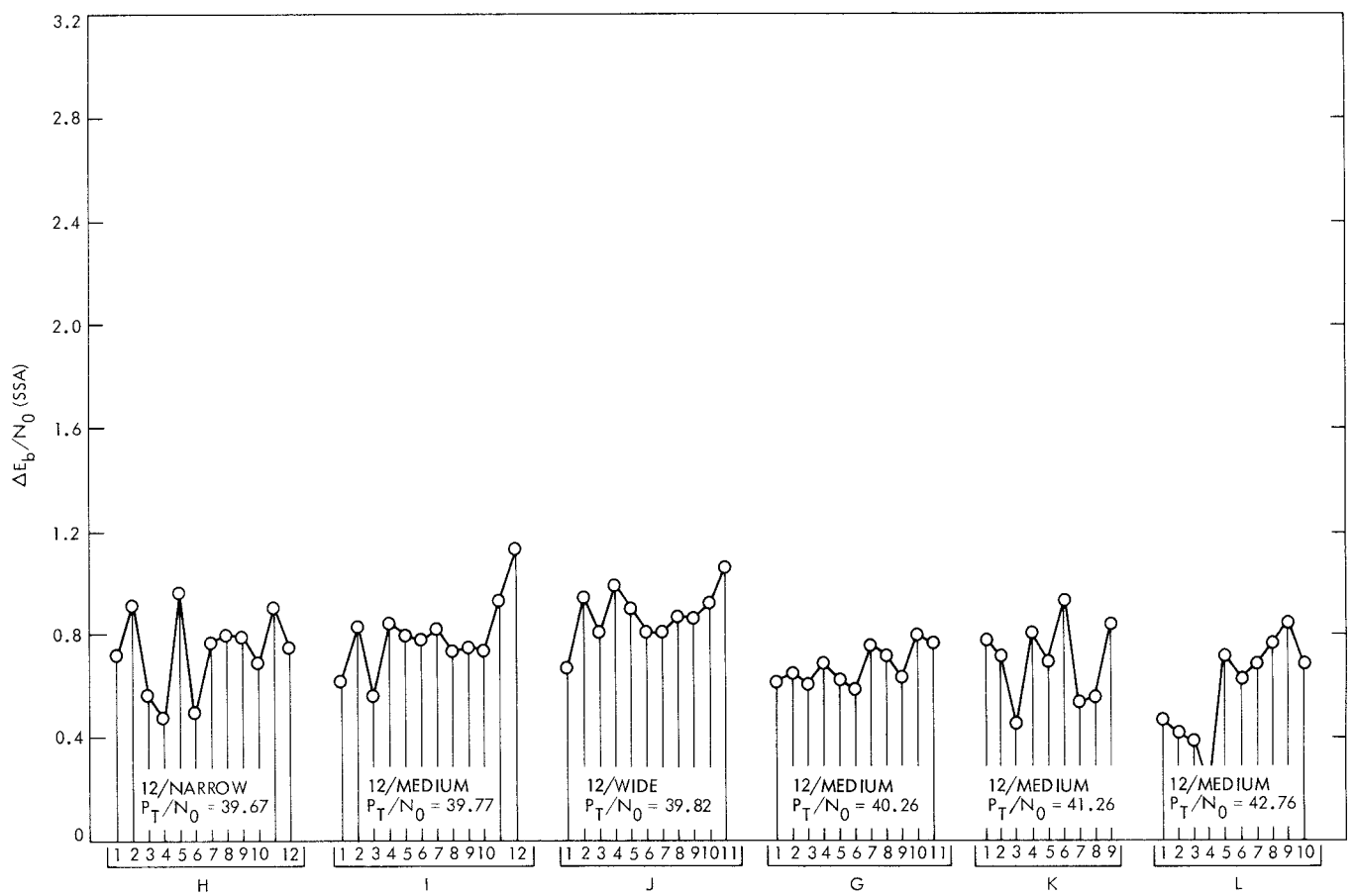

Fig. 3. Difference between E_b/N_o (Th) and E_b/N_o (SSA)

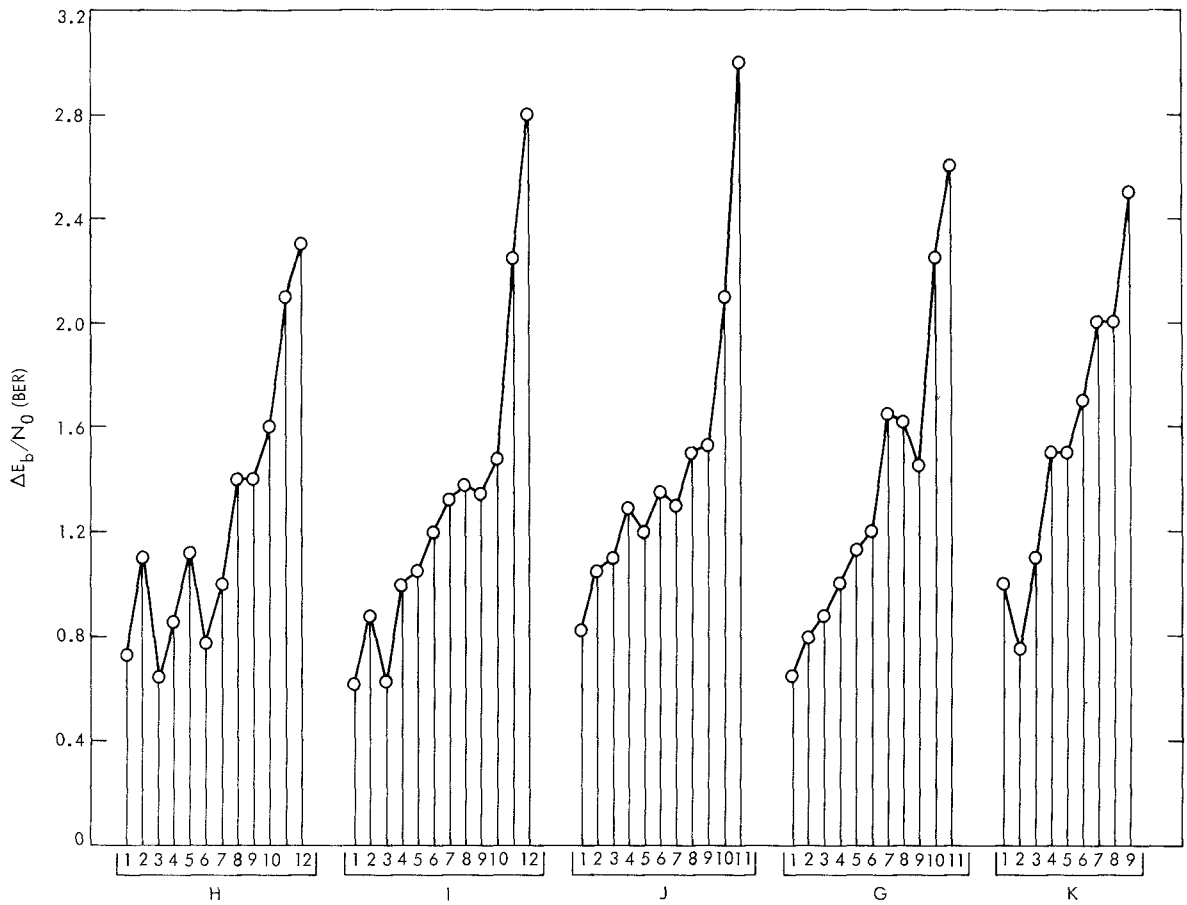

Fig. 4. Difference between E_b/N_o (Th) and E_b/N_o (BER)

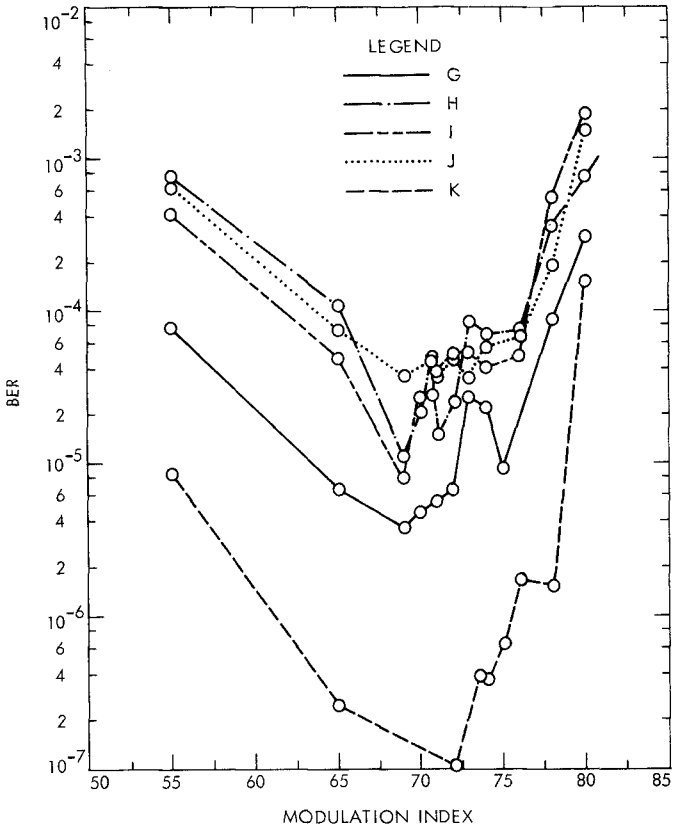

Fig. 5. Bit error rate versus modulation index

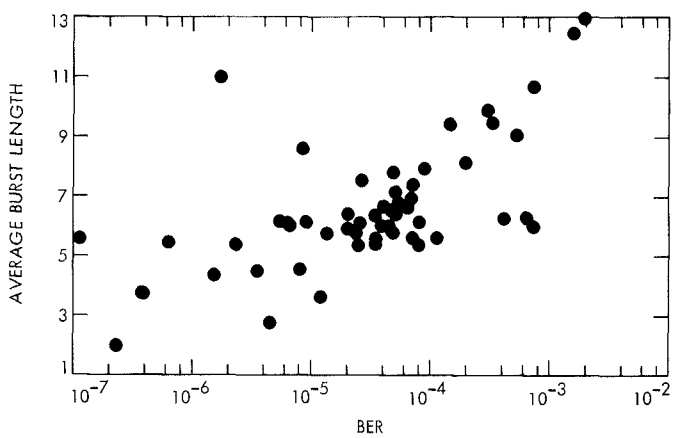

Fig. 6. Average burst length versus bit error rate

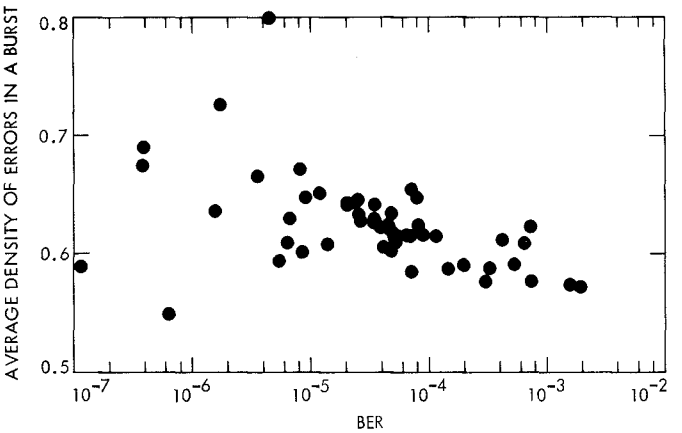

Fig. 7. Average density of errors in a burst versus bit error rate

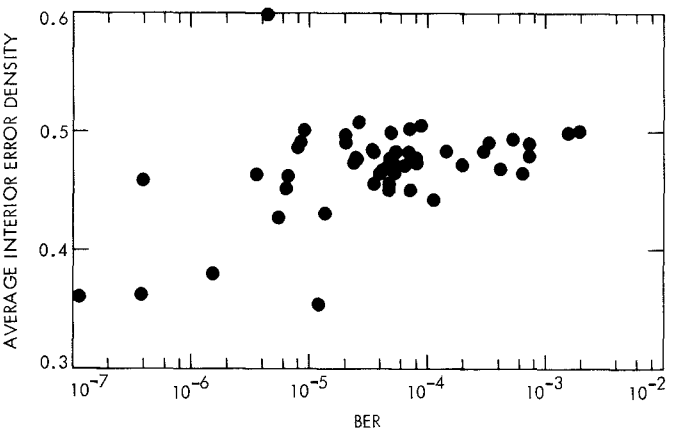

Fig. 8. Average interior error density versus bit error rate

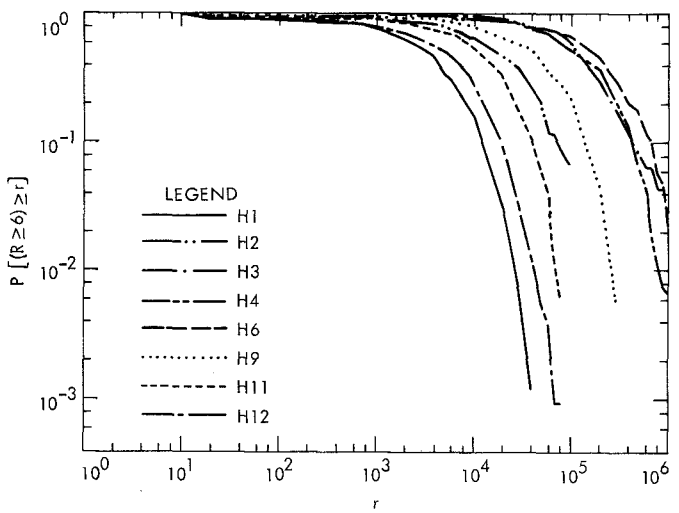

Fig. 9. Sample distribution functions of run length ($R \geq 6$)

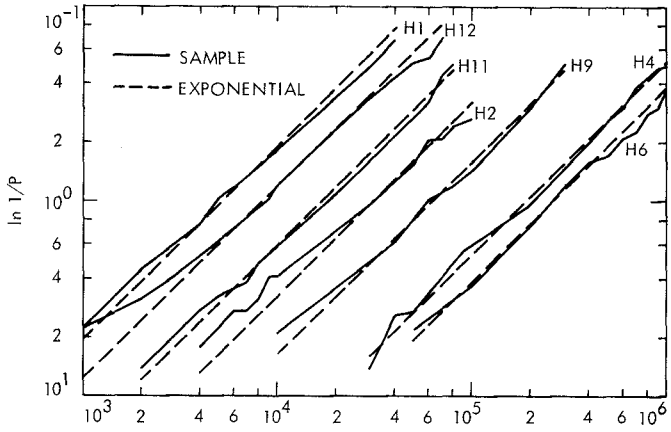

Fig. 10. Comparison of sample distributions of runs with exponential distributions

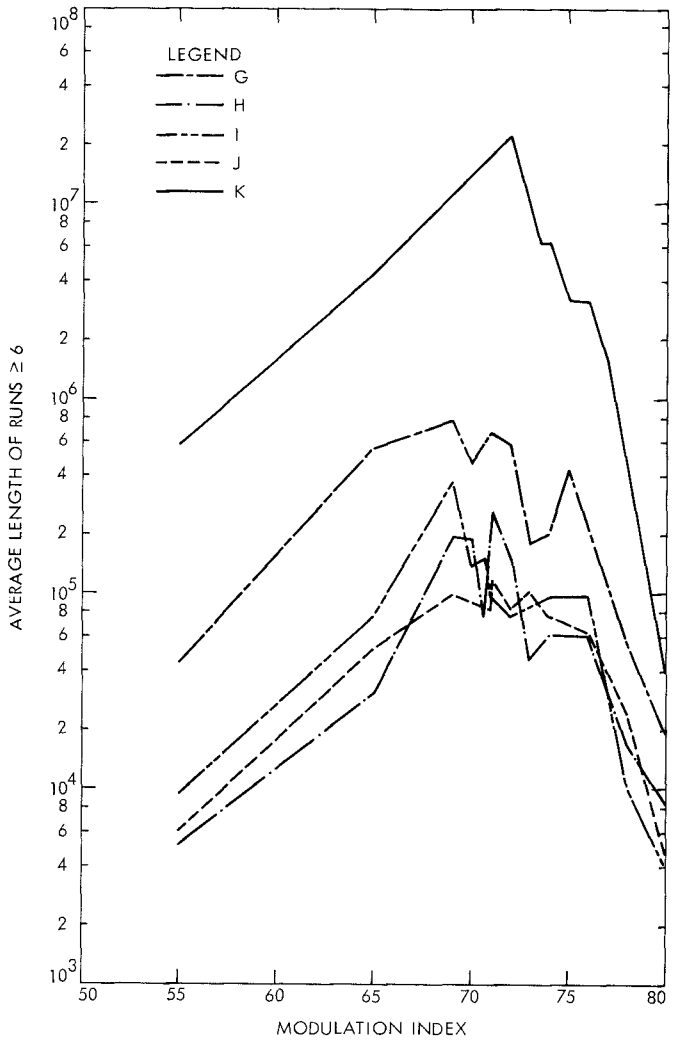

Fig. 11. Average length of runs ≥ 6 versus modulation index

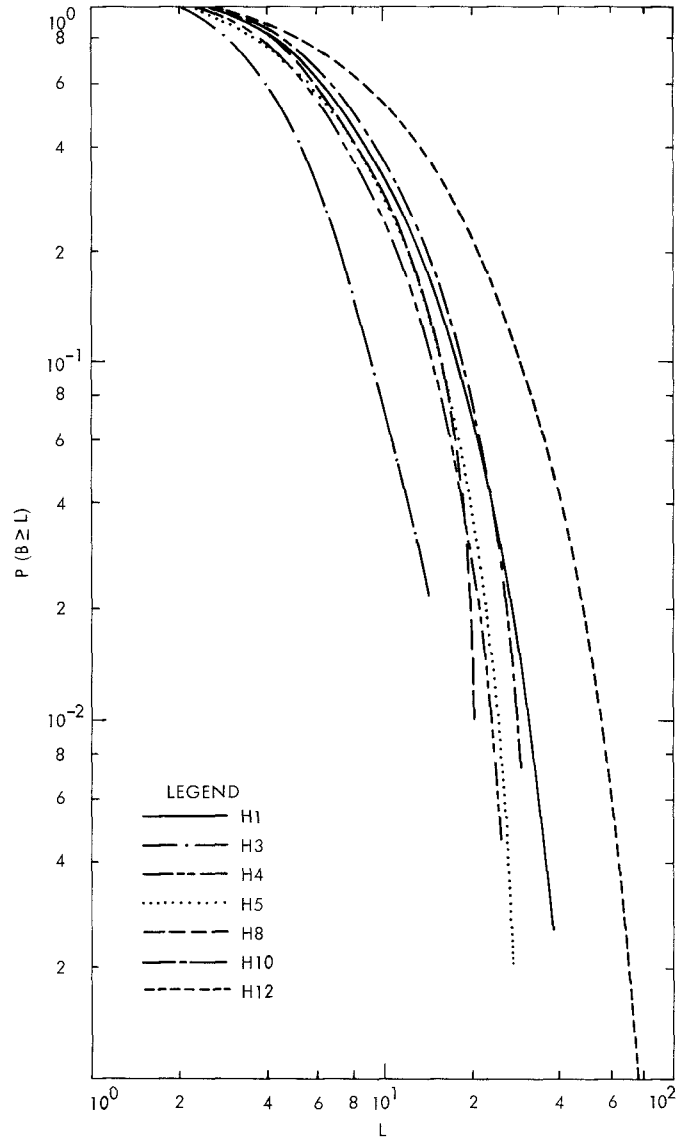

Fig. 12. Distribution functions of burst length

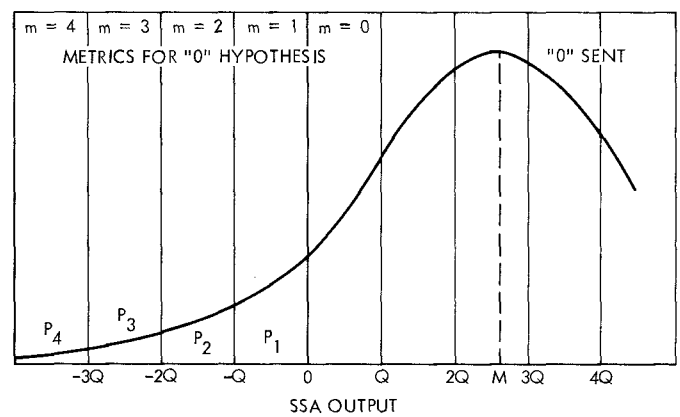

Fig. 13. Quantization and metric schemes

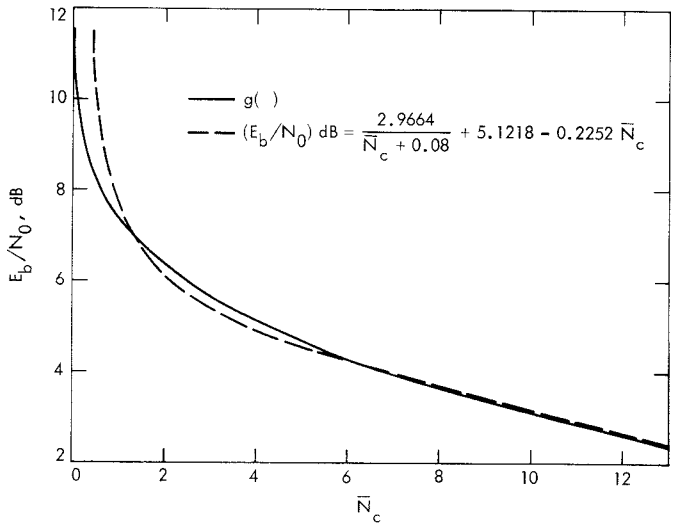

Fig. 14. E_b/N_o versus normalization counts

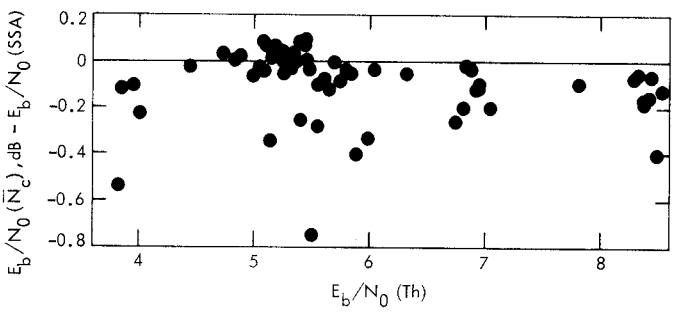

Fig. 15. Comparison of normalization and SAA estimates of E_b/N_o

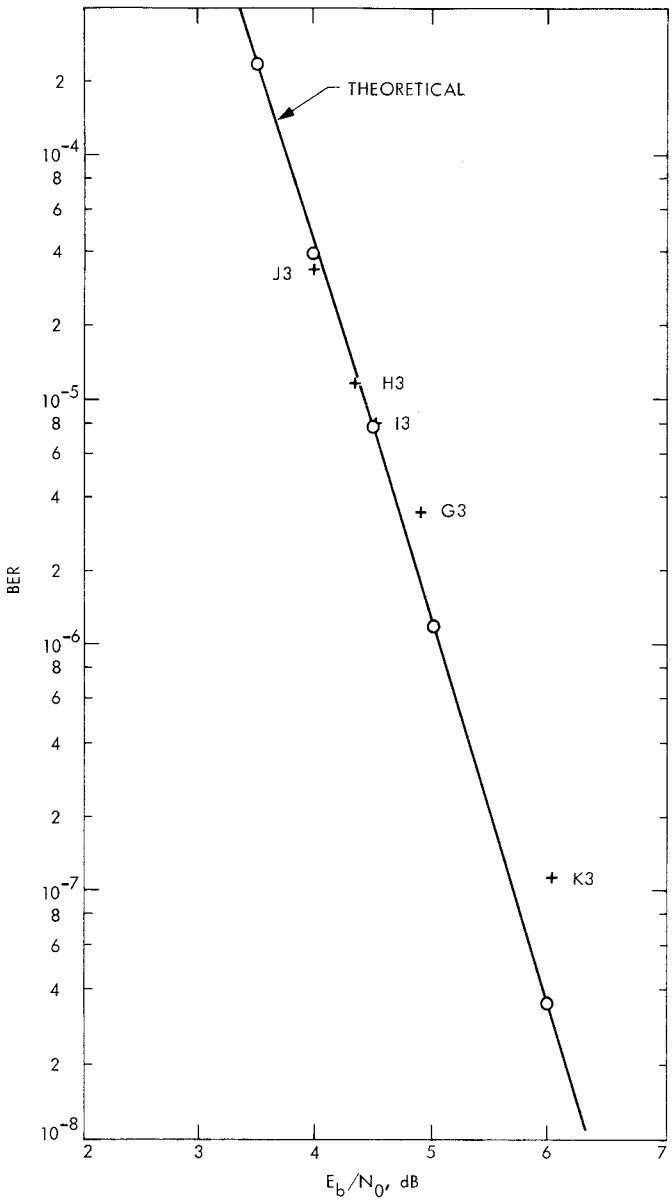

Fig. 16. Bit error rate versus E_b/N_o ($\bar{N}_c$) for runs having optimum modulation indexes