VLA Telemetry Performance with Concatenated Coding for Voyager at Neptune

S. J. Dolinar, Jr.
Communications Systems Research Section

Current plans for supporting Voyager's encounter at Neptune include the arraying of the DSN antennas at Goldstone, California, with the National Radio Astronomy Observatory's Very Large Array (VLA) in New Mexico. Not designed as a communications antenna, the VLA's signal transmission facility suffers a disadvantage in that the received signal is subjected to a "gap" or blackout period of approximately 1.6 msec once every 5/96 sec control cycle.

Previous analyses showed that the VLA data gaps could cause disastrous performance degradation in a VLA stand-alone system and modest degradation when the VLA is arrayed equally with Goldstone. These basic conclusions were independent of whether Voyager was using its convolutional code alone or the convolutional code concatenated with its Reed-Solomon outer code.

New analysis indicates that the earlier predictions for concatenated code performance were overly pessimistic for most combinations of system parameters, including those of Voyager-VLA. The periodicity of the VLA gap cycle tends to guarantee that all Reed-Solomon codewords will receive an average share of erroneous symbols from the gaps. The number of gapped symbols is not subject to the same kind of statistical fluctuations that govern the ordinary random errors the code must also overcome. However, large deterministic fluctuations in the number of gapped symbols from codeword to codeword may occur for certain combinations of code parameters, gap cycle parameters, and data rates. In this article, several mechanisms for causing these fluctuations are identified and analyzed.

Fortunately, the Voyager-VLA parameters do not produce wild fluctuations in the number of gapped symbols from codeword to codeword. The result is graceful degradation of concatenated code performance due to the VLA gaps, even for a VLA stand-alone system. The magnitude of the deterioration at a constant concatenated code bit error rate of 10^{-5} is 0.5 dB to 0.6 dB for a VLA stand-alone system and 0.3 dB to 0.4 dB for the VLA arrayed equally with Goldstone.

Even though graceful degradation is predicted for the Voyager-VLA parameters, catastrophic degradation greater than 2 dB can occur for a VLA stand-alone system at certain non-Voyager data rates inside the range of the actual Voyager rates. Thus, it is imperative that all of the Voyager-VLA parameters be very accurately known and precisely controlled.

I. Introduction

Current plans [1] for supporting Voyager's encounter at Neptune include the arraying of the DSN antennas at Goldstone, California, with the National Radio Astronomy Observatory's Very Large Array (VLA) in New Mexico. The fully arrayed VLA operating in a stand-alone mode potentially provides about the same receiving capability as the Goldstone complex. The VLA arrayed with Goldstone would seem to offer up to two times greater data rates than Goldstone alone.

Not designed as a communications antenna, the VLA's signal transmission facility unfortunately suffers a disadvantage in that the received signal is subjected to a "gap" or blackout period of approximately 1.6 msec once every control cycle. During the blackout period, the received signal is not transmitted from the antennas to the processing facility. The control cycle is 5/96 sec (approximately 52 msec), so the blackout period constitutes about 3% of the total receiving time.

If the VLA were used in a stand-alone mode to receive uncoded data, the data received during the gaps would be irretrievably lost. The resulting bit error rate, averaged over gapped and ungapped periods, could be no better than about 1.5%, even if no errors occurred outside the gaps. Arraying the VLA with Goldstone provides some capability during the gapped periods, but the overall error rate is still at least 3% of the error rate that would prevail based on the Goldstone-only aperture without any assistance from the VLA.

The high raw error rates during the gaps can potentially be overcome by coding the data. All of Voyager's telemetry data is convolutionally encoded, and the memory and error correction capability of the convolutional code provides a mechanism for bridging small gaps in the data. Unfortunately, the convolutional code's correction capability is limited to approximately the memory length of the code, and the VLA gaps are longer than the Voyager code's memory length (6 bits) for data rates greater than 3.75 kbits/sec.

Voyager's compressed imaging data is Reed-Solomon encoded in addition to being convolutionally encoded. Each Reed-Solomon codeword consists of 255 eight-bit symbols, and blocks of four codewords are interleaved symbol by symbol. Thus, more than 8000 data bits are transmitted between the beginning and end of a codeword. At Voyager-Neptune data rates of 21.6 kbits/sec or lower, each Reed-Solomon codeword is decoded based on symbols accumulated over a minimum of seven or eight complete gap cycles, so the Reed-Solomon decoder tends to see an average mix of gapped and ungapped symbols. Since Voyager's Reed-Solomon code can correct about 6% erroneous symbols, the code can potentially withstand 3% gapped symbols with a reserve error correction capability of 3% to handle normal ungapped symbol errors.

II. Previous Analysis

The general conclusions based on the simple reasoning in Section I are largely valid, but more detailed analysis is necessary to quantify the deleterious effects of the data gaps and to detect anomalous situations when the "average" behavior is not a valid determinant of overall performance. An analysis of the effects of the VLA data gaps on Voyager's convolutionally coded and concatenated coded data was reported years ago [2], [3] when the possible use of the VLA for Voyager was first foreseen. The earlier analysis first examined the effects of the data gaps on convolutionally coded data via a full software simulation of the Viterbi decoder that accurately modeled the VLA gap cycle. Then the average Viterbi decoder error rates predicted by the simulation were used as the basis for a theoretical calculation of the performance of the convolutional/Reed-Solomon concatenated code.

The intuitive conclusions about the performance of a stand-alone VLA receiving convolutionally coded data were borne out by the simulations. As shown in Fig. 4 of [3], the decoded error rate decreases slowly as a function of signal-to-noise ratio and then flattens out at an unacceptable value around 1%. The exact limiting error rate is a function of the data rate and the duty cycle of the gap period, and it approaches 1.5% for high data rates and the VLA duty cycle of 3% gaps. When the VLA is arrayed equally with Goldstone, so that the overall signal-to-noise ratio drops by only 3 dB during the gaps, the error rate curve retains its usual character and does not approach a saturation value over the interesting range of error rates.

The earlier theoretical calculations of the Reed-Solomon code's performance in Fig. 9 of [3] show the same leveling off of error rate as a function of signal-to-noise ratio when the VLA is unassisted by Goldstone. This implies unacceptable performance for a VLA stand-alone system, no matter how high the signal-to-noise ratio at the VLA. The Reed-Solomon performance deterioration when the VLA's signal is augmented by an equal signal from Goldstone is not so dramatic, as the error rate curves again retain their usual character but drop off more slowly.

The earlier analysis reached sharply different conclusions about VLA performance with and without assistance from Goldstone. Because of the predicted potential for devastating degradation, the causes of which were not fully understood, the VLA gap analysis was reopened in order to pin down the precise error mechanisms before Voyager's Neptune encounter

next year. It was not possible to improve on the earlier analysis of the convolutional code's performance, because the simulation accurately modeled both the Viterbi decoder and the VLA gap cycle. However, a somewhat more detailed analysis of the Reed-Solomon code's performance was undertaken, and this new analysis is reported here.

The new analysis of the performance of concatenated coding yields less pessimistic conclusions about the effect of the data gaps in a VLA stand-alone system. The regularity of the gap cycle helps to eliminate the possibility of larger than average numbers of errors due to the gaps. On the average, the Reed-Solomon code's error correction capability can take care of errors during the gaps, and this average behavior is overwhelmingly likely to occur for most combinations of system parameters. On the other hand, the new analysis reveals that certain combinations of parameters are taboo if the type of ruinous degradation predicted by the old analysis is to be avoided.

III. Various Possible Analytical Approaches

There are several possible approaches to calculating the theoretical performance of the Reed-Solomon code in the presence of data gaps. Three such approaches are shown in Fig. 1. The simplest such approach, called the one-level model, was the approach used in the earlier analysis. The most complex and accurate approach, the simulated error stream model, is not feasible. The middle approach, the two-level model, is the one taken in the current analysis.

The single-level model is based on the following expression (cf. Eq. 2 of [2]) for calculating the concatenated code's bit error probability:

$$P_b = \frac{p}{\pi} \sum_{i=E+1}^{N} \frac{i}{N} \binom{N}{i} \pi^i (1-\pi)^{N-i} \qquad (1)$$

Equation (1) expresses the bit error probability P_b of the concatenated channel in terms of the bit error rate p and the Reed-Solomon symbol error rate π of the output from the Viterbi decoder. This expression assumes an $(N, N - 2E)$ Reed-Solomon code, which can correct up to E symbol errors per N-symbol codeword. In [2] and [3], the average error probabilities p and π characterizing the Viterbi decoder output were obtained by means of a detailed simulation of the Viterbi decoding process, including an accurate model of the signal-to-noise ratio fluctuations over the VLA gap cycle. The model based on Eq. (1) is termed the "single-level" model, because the Reed-Solomon error probability is calculated using single overall average values of p and π to characterize the Viterbi decoder behavior, without regard to deterministic fluctuations in p and π between the gapped and ungapped portions of a gap cycle.

The validity of Eq. (1) rests on the assumptions that successive Reed-Solomon symbol errors are independent and identically distributed. As stated in [2], independence of symbol errors is a good assumption for Voyager because Voyager's 8-bit Reed-Solomon symbols are interleaved to depth 4 and Viterbi decoder error bursts of 32 or more bits are highly unlikely for the (7, 1/2) convolutional code. On the other hand, the assumption of identically distributed symbol errors throughout a Reed-Solomon codeword should be altered to account for the deterministic periodicity of the gap cycles. Symbols occurring during the gaps have a higher error rate than symbols occurring outside the gaps.

Ideally, a set of N values of p and π should be calculated from the Viterbi decoder simulation for each possible starting "phase" of the gap cycle relative to the Reed-Solomon codeword boundaries. Equation (1) can be easily modified to allow the values of p and π to vary symbol by symbol throughout the codeword. The error rate calculated from this "multilevel" model can then be averaged over all possible relative phases of the gap cycle to obtain the overall average bit error rate.

The analysis in the present article is not based on this general multilevel model for the Viterbi decoder's output statistics. Rather, it assumes that two levels will suffice: one set of values for p and π during the ungapped portion of the gap cycle and another set of values during the gaps. Intuitively, the two-level model should become exact in the limit of very high data rates, as the widths of both the gapped and ungapped periods become long with respect to the memory length of the convolutional code. In this limit, the Viterbi decoder has a chance to settle into steady-state values of p and π both inside and outside the gaps, and the number of "transition" bits and symbols characterized by intermediate values of ρ and π is small relative to the number of bits and symbols characterized by the steady-state gapped and ungapped values.

Another form of the multilevel model consists of dispensing with the theoretical calculations altogether and instead feeding the simulated output of the Viterbi decoder directly into a simulation of the Reed-Solomon code's performance. Tests of this type are impractical because of the monumental amount of simulated data that must be collected before statistical confidence in the results can be obtained. For example, at an operating point of 10^{-4} Reed-Solomon codeword error probability (corresponding to a concatenated code bit error probability of about 3×10^{-6}), the average waiting time for each erroneous codeword is about 20 million

bits. Simulating enough Viterbi decoded data to produce a statistically valid sample of erroneous codewords was not feasible. However, similar end-to-end tests of the gapped VLA data have been performed using real test data from CTA-21.[1]

IV. Details of the Two-Level Model

At Voyager's data rates, the length of the ungapped portion of each gap cycle is several hundred to more than a thousand bits long, so the steady-state assumption on which the two-level model is based appears justified for the ungapped zone. The length of the gaps, however, is at most around 35 bits (or about 6 memory lengths of the convolutional code) at Voyager's highest data rate of 21.6 kbits/sec. Thus the accuracy of the two-level model is somewhat questionable for the gapped zone. However, it should still give a better prediction of concatenated code performance than the single-level model.

The two-level model used in this article is a model for the decoded output of the Viterbi decoder. The Viterbi output bit error rate is allowed to vary between two levels. The two corresponding types of errors are referred to as "gapped" errors and "ungapped" errors, respectively. Each Viterbi decoded bit is characterized by one of two bit error probabilities p_0 or p_1, and each Reed-Solomon symbol is characterized by one of two symbol error probabilities π_0 or π_1. Gapped bits and symbols have error probabilities p_0 and π_0, and ungapped bits and symbols have error probabilities p_1 and π_1.

The ungapped error probabilities p_1 and π_1 are assumed to be the steady-state Viterbi decoder output error probabilities for a decoder operating at the ungapped signal-to-noise ratio E_b/N_0, and the gapped error probabilities p_0 and π_0 are assumed to be the corresponding error probabilities for a decoder operating at the reduced signal-to-noise ratio inside the gap. The signal-to-noise ratio inside the gap is zero for the VLA stand-alone system, and for the VLA arrayed with Goldstone it is reduced from the signal-to-noise ratio outside the gap by an amount reflecting the array ratio. For equal contributions from the VLA and Goldstone, the signal-to-noise ratio reduction inside the gaps is 3 dB. Other array ratios considered in this article correspond to gap reductions of 1.5 dB and 5 dB.

A summary of the two-level model for the cases of a VLA stand-alone system and the VLA arrayed with Goldstone is shown in Table 1. Performance curves showing the relationship between the bit and symbol error probabilities and the signal-to-noise ratio for the Voyager code parameters are shown in Fig. 2[2]. The curves in this figure are taken from Fig. 3-1(a) of [4], together with the results of some new simulations of the Voyager code for signal-to-noise ratios E_b/N_0 near 0 dB or lower. The Viterbi decoder's performance in this normally uninteresting range of E_b/N_0 is relevant for the VLA gap analysis, because signal-to-noise ratios inside the gap can be 0 dB or lower whenever signal-to-noise ratios outside the gap are near the normal operating point of the decoder.

The classification of bits and symbols as "gapped" or "ungapped" is not very precise. To account for the error correction capability of the Viterbi decoder, some of the bits decoded during the gap period should be considered to be effectively the same as ungapped bits. By an argument presented in [2], the Viterbi decoder can correct exactly $K-1$ of the gapped bits when the signal-to-noise ratio is zero inside the gaps and infinite outside the gaps. Here, K is the constraint length and $K-1$ is the memory length of the convolutional code ($K = 7$ for the Voyager code). In this limiting case, the effect of the convolutional code is equivalent to converting $K-1$ gapped bits (characterized by the signal-to-noise ratio inside the gap) into $K-1$ ungapped bits (characterized by the signal-to-noise ratio outside the gap). In this article, it is assumed that the convolutional code effectively accomplishes this same conversion of $K-1$ gapped bits into $K-1$ ungapped bits, even though the signal-to-noise ratios of interest are not exactly zero inside the gaps or infinite outside the gaps.

Blocks of J consecutive bits are grouped to form symbols for the Reed-Solomon code ($J = 8$ for the Voyager code). A J-bit Reed-Solomon symbol is in error if any one of its bits is incorrect. Therefore, it is appropriate to classify a Reed-Solomon symbol as a gapped symbol if at least one of its J component bits is a gapped bit. Because one gapped bit can cause a whole J-bit symbol to be classified as a gapped symbol, a block of consecutive gapped bits is effectively lengthened by an average of $J-1$ bits for the purpose of calculating the number of gapped symbols. In other words, a block of B consecutive gapped bits corresponds, on the average, to $(B+J-1)/J$ gapped symbols. This effect will be analyzed more closely in Section VII.

[1]M. Varuna, "VLA Standalone Test Results for 3.6 KBPS and 7.2 KBPS Voyager Telemetry Data Rates," Interoffice Memorandum Voyager-GDSE-87-056, Jet Propulsion Laboratory, Pasadena, California, September 16, 1987.

[2]The Viterbi decoder error probabilities in Fig. 2 are plotted versus the signal-to-noise ratio E_b/N_0 for the convolutional code only. In this figure, E_b represents the signal energy per convolutionally encoded bit. In Figs. 3 through 6 and 14 through 19, which show concatenated code error probabilities, the E_b/N_0 axis represents the signal-to-noise ratio for the concatenated system, i.e., E_b is the signal energy per Reed-Solomon encoded information bit.

The actual length of the gaps is effectively reduced by about $K - 1$ bits due to the error correction capability of the Viterbi decoder, but then increased by an average of $J - 1$ bits by the ability of one bad bit to knock out an entire symbol. The net adjustment, $J - K$, equals just one bit for the Voyager code parameters ($J = 8$, $K = 7$), so any reasoning based on the physical gap length rather than the effective gap length is probably valid for the Voyager case. However, the model used in this article will keep track of these two compensating effects separately, so it can be applicable to combinations of K and J which might not cancel each other so neatly.

Since the two-level model is a model for the Viterbi decoder output statistics, it can be tested against the results of the detailed simulations conducted in [2] and [3]. The test can check whether the overall Viterbi decoder error statistics (averaged over gapped and ungapped periods) predicted from the simulations match the average of the two levels of statistics used in the two-level model. The test cannot directly check whether the two-level model is adequate for the purposes of calculating concatenated code performance, since detailed Reed-Solomon code performance simulations coupled with the Viterbi decoder simulations are not available as a benchmark. A corroboration of the two-level model at the level of the Viterbi decoded output is reported in the Appendix.

V. Code Parameters, Gap Cycle Parameters, and Data Rates

In order to apply the two-level model to the calculation of the effects of the VLA gaps on concatenated code error rates, several additional parameters need to be defined. The constraint length K or memory length $K - 1$ of the convolutional code and the symbol size J of the Reed-Solomon code's symbols have already been discussed in the description of the basic model. Other code parameters that affect the concatenated code performance are the Reed-Solomon code's word length N, its error correction capability E, and its interleaving depth I. Essential parameters of the VLA gap cycle are the gap length G and the total length of the gap cycle T. The final parameter that influences the model's prediction of concatenated code performance is the data rate R. In this article, R is defined as the Viterbi decoder output bit rate. The corresponding channel symbol rate is $2R$ for Voyager's rate 1/2 convolutional code. The redundant nondata bits inserted by the Reed-Solomon code are counted toward the data rate R as defined here.

Table 2 lists the values of these essential parameters for the Voyager-VLA configuration.

VI. Conditional Concatenated Code Error Probabilities

The Reed-Solomon decoder error probability depends on the two symbol error probabilities π_0, π_1, and also on the number of symbols of both types within a Reed-Solomon codeword. Let n_0 denote the number of "gapped" symbols with error probability π_0, and n_1 the number of "ungapped" symbols with error probability π_1. The total Reed-Solomon word length is $N = n_0 + n_1$. The number of gapped symbols n_0 depends on the data rate R, the gap length G and gap period T, the Reed-Solomon symbol length J and codeword length N, the convolutional code constraint length K, and the "phase" of the gap cycle relative to the Reed-Solomon codeword boundaries.

The symbol error probability for the output of the Reed-Solomon decoder can be evaluated from a generalization of Eq. (1) that accounts for the two input symbol error probability levels. The answer also depends on whether it is evaluated for a gapped symbol or an ungapped symbol. If P_{s0} and P_{s1} denote the output gapped and ungapped symbol error probabilities, respectively, then

$$P_{s0} = \sum_{\substack{0 \le i \le n_0 \\ 0 \le j \le n_1 \\ i+j > E}} \sum \frac{i}{n_0} \binom{n_0}{i} \pi_0^i (1 - \pi_0)^{n_0 - i} \binom{n_1}{j} \pi_1^j (1 - \pi_1)^{n_1 - j} \tag{2}$$

$$P_{s1} = \sum_{\substack{0 \le i \le n_0 \\ 0 \le j \le n_1 \\ i+j > E}} \sum \frac{j}{n_1} \binom{n_0}{i} \pi_0^i (1 - \pi_0)^{n_0 - i} \binom{n_1}{j} \pi_1^j (1 - \pi_1)^{n_1 - j} \tag{3}$$

The corresponding output bit error probabilities P_{b0} and P_{b1} are obtained by multiplying these expressions by the conditional probability of a bit error, given a symbol error,

$$P_{b0} = \frac{p_0}{\pi_0} P_{s0} \tag{4}$$

$$P_{b1} = \frac{p_1}{\pi_1} P_{s1} \tag{5}$$

where p_0 and p_1 are the Viterbi decoder output bit error probabilities for gapped and ungapped bits, respectively.

The overall average symbol and bit error rates P_s and P_b output from the Reed-Solomon decoder are obtained by averaging the expressions for P_{s0}, P_{s1} and P_{b0}, P_{b1} over the n_0 gapped symbols and the n_1 ungapped symbols.

$$P_s = \frac{n_0}{N} P_{s0} + \frac{n_1}{N} P_{s1} \tag{6}$$

$$P_b = \frac{n_0}{N} P_{b0} + \frac{n_1}{N} P_{b1} \tag{7}$$

The bit and symbol error probability formulas given in Eqs. (2) through (7) can be regarded as expressions for the conditional bit error probability, given knowledge of the number of gapped symbols n_0 in a codeword. The computation of this conditional error probability is a convenient intermediate step toward the eventual computation of the unconditional error probability, because it separates the performance evaluation into two reasonably distinct parts. The first part shows how sensitive the performance is to variations in the number of gapped symbols per codeword, but it can be analyzed without reference to any peculiarities of the gapping mechanism which might cause the number of gapped symbols to vary from codeword to codeword. The evaluation of the conditional bit error probability can be performed independently of many of the parameters in the problem, including the data rate R, the gap length G, the gap cycle T, and the code's interleaving depth I. The second step in the overall performance evaluation is to evaluate the interplay of these remaining parameters in determining the critical number of gapped symbols per codeword.

This separability of the overall problem, with just one crucial parameter serving to link the two parts of the analysis, is a big advantage in favor of the two-level model relative to multilevel models or a full-scale combined simulation of the Viterbi decoder and Reed-Solomon performance. Whatever exactness the simpler two-level model might lack relative to multilevel models is compensated by increased insight into what effect each of the parameters has on the overall performance.

Figures 3 through 6 show the evaluation of the Reed-Solomon conditional bit error probability for various values of the parameter n_0. Figure 3 shows the performance curves for a stand-alone VLA system, and the next three figures apply to a VLA-Goldstone array with varying contributions from each component of the array. The curves in each figure were evaluated using values of gapped and ungapped bit and symbol error probabilities, p_0, π_0, p_1, π_1, obtained from the baseline steady-state Viterbi decoder performance curves in Fig. 2. The Reed-Solomon code parameters N and E were fixed at the Voyager code's characteristics, $N = 255$ and $E = 16$.

It is instructive to examine the curves for the VLA stand-alone case. When the VLA is unassisted by Goldstone, the received data signals are totally lost during the time of the VLA gap. A total gap is characterized by zero signal-to-noise ratio and totally random steady-state Viterbi decoded bits. The gapped bit and symbol error probabilities are $p_0 = 1/2$ and $\pi_0 = (2^J - 1)/2^J$, where J is the number of bits comprising a symbol. Since $J = 8$ for the Voyager Reed-Solomon code, a good simplifying approximation is $\pi_0 \approx 1$. Substituting $\pi_0 = 1$ into Eqs. (2) and (3) leads to

$$P_{s0} = \sum_{j=E-n_0+1}^{N-n_0} \binom{N-n_0}{j} \pi_1^j (1-\pi_1)^{N-n_0-j} \tag{8}$$

$$P_{s1} = \sum_{j=E-n_0+1}^{N-n_0} \frac{j}{N-n_0} \binom{N-n_0}{j} \pi_1^j (1-\pi_1)^{N-n_0-j} \tag{9}$$

as long as n_0 is not greater than E. The error probability formula of Eq. (9) for the ungapped symbols is the same equation as for the symbol error probability under a one-level model for a code of blocklength $N - n_0$ capable of correcting $E - n_0$ errors. Thus, the effect of a total gap on the ungapped symbols is simply to use up some of the error-correcting capability of the Reed-Solomon code. The performance degradation in the ungapped zone due to the gap is equivalent to the degradation that would result from substituting a less powerful code. Figure 3 effectively shows how well a series of less and less powerful codes performs relative to the baseline performance of the Voyager code (corresponding to the curve in Fig. 3 labeled $n_0 = 0$). The performance curves deteriorate very rapidly as n_0 approaches 16, which represents a blocklength 239 code with no error correction capability. Values of n_0 greater than 16 would completely overwhelm the code.

The notion of an "equivalent" reduced-redundancy code for determining performance in the case of a total gap is useful but not completely accurate. The concatenated code's overall error statistics depend on the statistics for both ungapped symbols and gapped symbols. The error probability formula of Eq. (8) for gapped symbols equals the "equivalent" reduced-redundancy code's word error probability, which is greater than its symbol error probability. Thus, the performance degradation in the gapped zone due to the gap is worse than the degradation resulting from substituting the less powerful code. The overall average symbol error probability P_s is obtained by averaging the results of Eqs. (8) and (9) over gapped and ungapped symbols, as in Eq. (6), and hence is somewhat higher than the symbol error probability of the intuitively "equivalent" reduced-redundancy code. A similar conclusion holds for the overall average concatenated code bit error probability P_b, obtained from Eqs. (8) and (9) via Eqs. (4), (5), and (7).

VII. The Number of Gapped Symbols per Codeword

The number of bits decoded by the Viterbi decoder during each gap period is RG. The total number of Viterbi decoded bits in an entire gap cycle is RT. This corresponds to RG/r channel symbols that are received during gaps, out of RT/r channel symbols received every gap cycle if the convolutional code's rate is r ($r = 1/2$ for the Voyager code). Under the two-level model, approximately $K - 1$ (the memory length of the convolutional code) of the decoded bits during the gap period can be treated as having the same signal-to-noise ratio as ungapped bits. Thus, the effective length of each gap is reduced from RG bits to approximately $RG - K + 1$ bits due to the correction capability of the Viterbi decoder. On the other hand, the gap period is effectively lengthened by $J - 1$ bits, on the average, due to the ability of just one incorrect bit to corrupt an entire J-bit Reed-Solomon symbol. Thus, under the two-level model, the average number of gapped symbols in each N-symbol codeword is $N(RG - K + J)/RT$. This works out to an average of about eight gapped symbols per 255-symbol codeword for the Voyager-VLA parameters listed in Table 2.

The periodicity of the gap cycle tends to guarantee that every Reed-Solomon codeword receives an average number of gapped symbols. However, there are certain conditions under which this conclusion is invalid. Some codewords can get more than their share of gapped symbols, while other codewords receive fewer. The codewords receiving too many gapped symbols are drastically more prone to error, as indicated by the rapid deterioration in the performance curves in Figs. 3 to 6 as the number of gapped symbols n_0 is increased. Thus, a small number of such atypical codewords can dominate the overall error performance of the concatenated code.

A. Fluctuations Due to Symbol Edge Effects

One basic mechanism causing an uneven distribution of gapped symbols per codeword is the symbol edge effects that on the average lengthen the effective gap by $J - 1$ bits. The actual lengthening of the gap can vary from 0 bits to $2J - 2$ bits, depending on the "phase" of the gap edges relative to symbol boundaries. Figure 7 shows that each gap is lengthened by exactly $(\phi_1 + \phi_2)$ bits, where ϕ_1 and ϕ_2 are the phases of the left and right edges of the effective gap relative to Reed-Solomon symbol boundaries. Both of these phases are uniformly distributed from 0 to $J - 1$ bits for a codeword picked at random. However, the two phases are not independent of each other, and in fact they must satisfy $[(\phi_1 + \phi_2) + (RG - K + 1)] \bmod J = 0$. Despite this dependence, the average of $(\phi_1 + \phi_2)$ is always $J - 1$ bits, because the average of a sum of random variables equals the sum of the averages even when the random variables are correlated. In general, $(\phi_1 + \phi_2)$ can assume either of two values, except that when $(RG - K + 1) \bmod J = 1$, it must equal $J - 1$ bits regardless of where the symbol boundaries fall relative to the gap edges. When $(RG - K + 1) \bmod J \neq 1$, the two possible values for $(\phi_1 + \phi_2)$ are separated by exactly J bits. Thus, the actual lengthening of the gap due to symbol edge effects can be one of two values which differ by one symbol.

For example, when $(RG - K + 1) \bmod J = 0$, the two possible values for $(\phi_1 + \phi_2)$ are 0 bits and J bits, and when $(RG - K + 1) \bmod J = 2$, the two possible values are $J - 2$ bits and $2J - 2$ bits. In the first example, $(\phi_1 + \phi_2) = 0$ with probability $1/J$ and $(\phi_1 + \phi_2) = J$ with probability $1 - 1/J$. In the second example, $(\phi_1 + \phi_2) = J - 2$ with probability $1 - 1/J$ and $(\phi_1 + \phi_2) = 2J - 2$ with probability $1/J$. In both cases, the average value of $(\phi_1 + \phi_2)$ is $J - 1$ bits. However, the conditions in the second example will cause slightly poorer concatenated code performance, because the worst-case lengthening of the gap is $J - 1$ bits greater than the average lengthening. On the other hand, the performance curves for the optimal data rates, which result in $(RG - K + 1) \bmod J = 1$, will suffer no additional degradation beyond that due to the average lengthening of the gaps.

The additional degradation due to fluctuations in the lengthening of the gaps as a result of symbol edge effects is depicted in Fig. 8 as a function of the data rate R. In this figure the extra degradation is measured in terms of the worst-case number of gapped symbols per gap relative to the average number. The worst-case number of gapped symbols per gap varies periodically with the data rate between a minimum value of $(RG - K + J)/J$ gapped symbols and a maximum value of $(RG - K + J)/J + 1 - 1/J$ gapped symbols. The period of this variation is J/G. For the Voyager-VLA parameters, the worst data rates occur nominally at 5 kbits/sec, 10 kbits/sec, 15 kbits/sec, . . . , and the best data rates occur nominally at 4.375 kbits/sec, 9.375 kbits/sec, 14.375 kbits/sec, ... The exact locations of the best data rates or the worst data rates are determined under the two-level model not only by the precisely measurable value of G, but also by the assumed effective shortening of each gap by exactly $K - 1$ bits due to the error correction capability of the convolutional code. Since the effective shortening of the gap is a fuzzy quantity, the absolute locations of the best or worst data rates cannot be determined precisely. Fortunately, the variation between the best and worst data rates is relatively small, because it is equivalent to creating less than one additional gapped symbol per gap. Other mechanisms causing fluctuations in the number of gapped symbols per codeword can cause much larger effects on performance.

B. Fluctuations Due to Incomplete Gap Cycles per Codeword

A second mechanism that could cause some codewords to have an atypically large number of gapped symbols occurs when the total span of one block of interleaved codewords encompasses one more gapped section of data than another block of interleaved codewords. One block of I interleaved codewords, each consisting of N J-bit symbols, spans a continuous section of NIJ bits. If NIJ is an exact integer multiple of the gap cycle RT, than all interleaved codeword sets will include gapped symbols from exactly NIJ/RT different gap cycles. However, if NIJ/RT is not an integer, the span of an interleaved set of codewords will include a number of complete gap cycles plus a fraction of a cycle. If the extra fractional cycle includes the gap, the interleaved codeword set is "unlucky" and will suffer degraded performance, because its overall fraction of gapped symbols is larger than the nominal value of $(RG - K + J)/RT$. Other interleaved codeword sets are "lucky" and avoid the gap altogether in the fractional gap cycle, resulting in better performance than the nominal prediction. However, the overall unconditional concatenated code performance is dominated by the performance of the unlucky codeword sets, and so it is important to quantify how unlucky they can be.

Figure 9 illustrates the effects of incomplete gap cycles per interleaved codeword block. The upper picture shows the case of an unlucky codeword set with a fractional gap cycle that includes a gap, and the lower picture shows a lucky codeword set whose fractional gap cycle misses the gap. In general, an "average" interleaved codeword block includes NIJ/RT gapped sections of data, but a lucky codeword block includes only $\lfloor NIJ/RT \rfloor$, while an unlucky codeword block includes $\lfloor NIJ/RT \rfloor + 1$, where $\lfloor x \rfloor$ represents the largest integer not exceeding x. Each gapped section of data includes, on the average, $(RG - K + J)/J$ gapped symbols, which are distributed over the I interleaved codewords. The difference between the lucky codeword blocks and the unlucky ones is $(RG - K + J)/IJ$ gapped symbols per codeword per gap. The difference between an unlucky codeword block and an average one lies linearly (as a function of data rate) between 0 and $(RG - K + J)/IJ$ gapped symbols per codeword per gap, depending on how close NIJ/RT is to $\lfloor NIJ/RT \rfloor$ or to $\lfloor NIJ/RT \rfloor + 1$.

Figure 10 plots the peak-to-average concentration of gapped symbols due to incomplete gap cycles versus the data rate. The peak-to-average concentration of gapped symbols in the unlucky codewords varies between 1 and $1 + RT/NIJ$ as NIJ/RT varies between successive integer values. The concentration factor returns to 1 periodically at reciprocal data rates separated by T/NIJ, but it rises to increasing maximum values between its returns to 1. For the Voyager-VLA parameters, the maximum value of RT/NIJ is less than 1/7 even at the maximum Voyager-Neptune data rate of 21.6 kbits/sec. Thus, the overall magnitude of the degradation caused by incomplete gap cycles is limited to about one additional gapped symbol per codeword for Voyager. However, the effect of incomplete gap cycles can be very severe at higher data rates, namely data rates approaching NIJ/T (= 157 kbits/sec) or higher.

C. Fluctuations Due to Gap Cycle/Interleaving Cycle "Resonances"

A third mechanism that may cause an atypical concentration of gapped symbols in some codewords is possible "resonances" between the gap cycle and the interleaving cycle. Even if every interleaved codeword block were to experience exactly the same proportion of gapped and ungapped periods, there can be a worst-case codeword within the interleaved block which gets more than its share of gaps. In the worst conceivable case, one unlucky codeword might receive all of the interleaved block's gapped symbols, while the other $I - 1$ codewords escape with no gapped symbols at all. This would result in a worst-case peak-to-average concentration of gapped symbols in one unlucky codeword by a factor of I.

Some potential situations that may cause a concentration of gapped symbols in one unlucky codeword are illustrated in Fig. 11. In Fig. 11(a), the average effective gap period $RG - K + J$ is small enough to fit within one symbol period J, and the distance between successive gap periods (the gap cycle RT) is exactly an integer multiple of the interleaving cycle IJ. In this case, whichever of the I interleaved codewords includes the gap in its first symbol will also include all of the gaps contained within the entire span (NIJ bits) of the interleaved block. This unlucky codeword will receive I times its average share of gapped symbols, and the remaining $I - 1$ codewords will have only ungapped symbols.

Figure 11(b) illustrates a slightly different situation in which the effective gap period is still small, but the gap cycle RT is increased by enough to retard the occurrence of successive gaps by one symbol. In this case, successive gaps hit consecutive codewords, and all I codewords within the interleaved block receive a proportionate share of gapped symbols. Figure 11(c) illustrates a case in which the gap cycle RT is made slightly longer, such that it retards the occurrence of successive gaps by two symbols instead of one. Now, if the interleaving depth I is an even number, half of the interleaved codewords will get all of the gapped symbols, and the unlucky codewords will experience a peak-to-average concentration factor of 2. On the other hand, if the interleaving depth I is odd, the gaps will be distributed over all of the codewords in the interleaved block.

Figure 11(a) identifies a series of data rates R which cause major resonances between the gap cycle and the interleaving cycle. Specifically, the major resonances occur for values of R satisfying $RT \bmod IJ = 0$. Figure 11(c) shows that minor resonances can also occur when $RT \bmod IJ \neq 0$, if $(RT \bmod IJ)/J$ and I contain a common integer factor. At the major resonances, the peak-to-average concentration factor is I, while at the minor resonances the concentration factor is equal to the common integer factor of I and $(RT \bmod IJ)/J$.

The preceding conclusions about the magnitude of the peak-to-average concentration factor at the major and minor resonances are valid only for the types of cases depicted in Fig. 11, for which the gapped portion $RG - K + J$ of the total gap cycle RT is small enough to fit within one codeword symbol. Two other possible cases are illustrated in Fig. 12. In Fig. 12(a), the effective gap length $RG - K + J$ encompasses exactly two symbols, and the data rate is chosen to cause a major resonance between the gap cycle and the interleaving cycle (i.e., $RT \bmod IJ = 0$). In this case, the same two codewords are always hit by successive gaps, while the remaining $I - 2$ codewords escape the gaps altogether. The resulting peak-to-average concentration factor in this case is $I/2$.

Figure 12(b) illustrates a situation in which the effective gap length $(RG - K + J)$ is longer than one block of I interleaved symbols. The portion of the gap covering an integer multiple of IJ bits affects each of the I codewords equally, but the remaining portion of the gap covering a fraction of IJ bits afflicts one or more of the I codewords selectively. If the data rate is at a major resonance, the same codeword(s) will remain unlucky for all the gaps that occur throughout the entire span of the interleaved codeword set. The unlucky codeword(s) will receive an extra share of gapped symbols corresponding to the fractional portion of gapped bits, namely $(RG - K + J) \bmod IJ$. An average codeword should receive $(RG - K + J)/IJ$ gapped symbols from each gap, but the lucky codewords receive only $\lfloor (RG - K + J)/IJ \rfloor$, while the unlucky codewords receive $\lfloor (RG - K + J)/IJ \rfloor + 1$ (unless $(RG - K + J)$ is exactly an integer multiple of IJ). The resulting peak-to-average concentration of gapped symbols in the unlucky codewords varies between 1 and $1 + IJ/(RG - K + J)$ as $(RG - K + J)$ varies between successive integer multiples of IJ.

The location of the major resonances depends on the length of the full gap cycle RT relative to the interleaving cycle IJ, while the magnitude of the performance deterioration at each major resonance depends on the effective length $(RG - K + J)$ of the gapped portion of a gap cycle relative to the interleaving cycle IJ. These two effects are depicted separately in Fig. 13. The variation of the peak-to-average concentration of gapped symbols is shown as a smooth curve, while the resonance locations are shown as sharp lines. The peak-to-average concentrations depicted by the smooth curve are valid only at or near the resonance locations. The resonances themselves are of nonzero width, but they are very narrow if the number of symbols N per codeword is large.

Peak-to-average concentration factors of 1.5 and greater are common at major resonances within the range of Voyager's data rates. The maximum peak-to-average concentration factor varies between 4 and 2 for data rates between 4.375 and 19.375 kbits/sec. A peak-to-average concentration factor of 1.5 corresponds to 12 gapped symbols per worst-case codeword rather than the average of 8. Figure 3 shows the potential for catastrophic performance degradation of a VLA standalone system when the number of gapped symbols per codeword reaches 12 or 16 or higher. Thus, it is important that the precise Voyager data rates miss the location of the narrow resonances. Fortunately, this is the case. Table 3 lists the data rates causing major resonances in the range from 3.75 to 22.5 kbits/sec. Even though one of the Voyager data rates (21.6 kbits/sec) appears to fall perilously close to a major resonance, the unconditional performance evaluations in the next section show that the small separation is sufficient to avoid resonant degradation.

VIII. Unconditional Concatenated Code Error Probability Curves for the Voyager-VLA Parameters

The number of gapped symbols per codeword, n_0, was evaluated as a function of the relative phase between the gap cycle and the codeword, assuming the Voyager-VLA parameter values listed in Table 2. This evaluation simultaneously takes into account all of the types of fluctuations identified in the previous section. It was found that the Voyager data rates are all nonresonant, in the sense that the worst-case value of n_0 was 9 or 10 and not 12, 16, or higher.

The overall unconditional concatenated code bit error rate is obtained by averaging the bit error rate in Eq. (7),

$$\bar{P}_b = \mathbf{E}\{P_b\} \tag{10}$$

where $\mathbf{E}\{\cdot\}$ represents an average over the possible values of n_0. Unconditional concatenated code bit error rates calculated from Eq. (10) are plotted in Figs. 14 through 17 for the same four VLA-Goldstone array ratios considered in Figs. 3 through 6.

The concatenated code performance curves in Figs. 14 through 17 are virtually identical for the three Voyager data rates shown, R = 7.2 kbits/sec, 14.4 kbits/sec, and 21.6 kbits/sec. Performance is very slightly improved at the higher rates. This negligible difference is attributable to the net effective

lengthening of the gap portion of the gap cycle by $J - K = 1$ bit. This net lengthening constitutes a larger fraction of the total gap cycle at lower data rates than at higher rates, and hence the gap is predicted to affect the lower data rates slightly more adversely. However, as pointed out earlier, the net effective lengthening of the gap is the result of two compensating effects which almost cancel each other. The convolutional code's error correction capability makes the gaps look shorter at the low data rates, while edge effects due to entire Reed-Solomon symbols getting wiped out by one erroneous bit make the gaps look shorter at the high data rates. The model for both of these effects is fuzzy enough that the predicted tiny performance improvement with increasing data rate is not significant. A more appropriate conclusion is that the predicted concatenated code performance is virtually independent of the Voyager data rate over the range $R = 7.2$ to 21.6 kbits/sec.

The performance curves in Figs. 14 through 17 are essentially identical to the $n_0 = 9$ conditional error rate curves in Figs. 3 through 6. This indicates that the unconditional error rate is almost completely determined by the error rate for codewords with the worst-case number of gapped symbols. At a constant performance level of 10^{-5} bit error rate, the net effect of the VLA gaps is to require 0.5 dB to 0.6 dB more signal-to-noise ratio for the VLA stand-alone system relative to an ungapped system. The net cost of the gaps when the VLA is arrayed equally with Goldstone (3-dB gaps) is 0.3 dB to 0.4 dB. When the VLA's contribution to the array is about twice Goldstone's (5-dB gaps), the net cost of the gaps is almost the same as for the VLA stand-alone system, 0.5 dB. When Goldstone's contribution is about twice the VLA's (1.5-dB gaps), the net cost of the gaps shrinks to 0.1 dB to 0.2 dB.

The modest amount of deterioration in the concatenated code performance due to the VLA gaps is a consequence of the reserve error correction capability of the Reed-Solomon code relative to the average number of gapped symbols, coupled with the fortuitous choice of nonresonant data rates for Voyager. A remarkable example of catastrophic performance degradation due to a resonant data rate is shown in Fig. 18. The resonant data rate, $R = 21.504$ kbits/sec, differs from one of the important Voyager rates by less than 0.5%, and yet this case suffers an additional performance degradation of around 2 dB for the VLA stand-alone system. The explanation is that 21.504 kbits/sec is a resonant data rate with a worst-case number of gapped symbols per codeword equal to 16 (see Table 3), which exhausts the error correction capacity of the Reed-Solomon code.

The effects of a resonant data rate are not so pronounced when the VLA is arrayed with Goldstone. Figure 19 shows concatenated code performance for the same resonant data rate considered in Fig. 18, but for the case of an equal VLA-Goldstone array ratio (3-dB gaps). The resonant data rate performance is only 0.1 dB to 0.2 dB worse than the performance for the nonresonant Voyager data rate at a required bit error probability of 10^{-5}. In fact, the concatenated code's performance at high bit error rates ($>10^{-3}$) is slightly better at the resonant rate than at the nonresonant rate. The reason for the dramatically improved performance is that, even though the Viterbi decoder's error rate during the 3-dB gaps is truly bad, it does not decode completely random bits as it does when the gaps are totally devoid of received data. If the Viterbi decoder manages to decode a few gapped symbols correctly, every correctly decoded gapped symbol adds one symbol's worth of reserve correction capacity to a Reed-Solomon decoder that would otherwise be operating with essentially no reserve capacity at all.

The main lesson to be drawn from Figs. 14 through 19 is that the Voyager data rates and the VLA gap cycle parameters must be very accurately known and precisely controlled in order to avoid a disastrous resonance that would ruin the performance of a VLA stand-alone system. If this can be done, the overall concatenated code performance degradation due to the VLA gaps can be limited to about 0.5 to 0.6 dB. When the VLA and Goldstone are arrayed in equal ratio, the nominal degradation is reduced to 0.3 to 0.4 dB, but, just as significantly, the extra degradation at a resonant data rate is only a few more tenths of a dB. Thus, the necessity to avoid a resonant data rate is not quite so critical if the VLA is arrayed with Goldstone.

IX. Summary

Voyager's Reed-Solomon outer code has sufficient error correction capacity to withstand the average number of erroneous symbols caused by the VLA data gaps. Of course, the code's reserve capacity for correcting ordinary random errors not caused by the gaps is diminished and the overall concatenated code performance is slightly degraded relative to that of an ungapped receiving system.

The periodicity of the VLA gap cycle tends to distribute an average number of gapped symbols to every Reed-Solomon codeword. However, several mechanisms were identified which can cause the actual number of gapped symbols to deviate from its benign average value for some unlucky codewords. These fluctuations are important because the overall performance of the concatenated code is dominated by its performance for the unluckiest codewords, which receive the worst-case number of gapped symbols. The mechanism causing the most serious fluctuations within the range of the Voyager-

VLA parameters is resonances between the VLA gap cycle and the Reed-Solomon codeword interleaving cycle. At many resonances within the range of Voyager's data rates, the number of gapped symbols included in unlucky codewords is 1.5 to 4 times higher than the average number. The performance degradation at these resonant data rates can be catastrophic, especially for a VLA stand-alone system, as seen in Fig. 18. Fortunately, the resonances are very narrow and none of the actual Voyager data rates falls disastrously near a resonant rate. However, the existence of these catastrophic resonant rates inside the range of the actual Voyager data rates underscores the importance of accurately knowing and precisely controlling all of the relevant code parameters, gap cycle parameters, and data rates for the Voyager-VLA system.

References

[1] J. W. Layland and D. W. Brown, "Planning for VLA/DSN Arrayed Support to the Voyager at Neptune," *TDA Progress Report 42-82*, vol. April–June 1985, pp. 125–135, Jet Propulsion Laboratory, Pasadena, California, August 15, 1985.

[2] L. J. Deutsch, "The Performance of VLA as a Telemetry Receiver for Voyager Planetary Encounters," *TDA Progress Report 42-71*, vol. July–September 1982, pp. 27–39, Jet Propulsion Laboratory, Pasadena, California, November 14, 1982.

[3] L. J. Deutsch, "An Update on the Use of the VLA for Telemetry Reception," *TDA Progress Report 42-72*, vol. October–December 1982, pp. 51–60, Jet Propulsion Laboratory, Pasadena, California, February 15, 1983.

[4] R. L. Miller, L. J. Deutsch, and S. A. Butman, *On the Error Statistics of Viterbi Decoding and the Performance of Concatenated Codes*, JPL Publication 81-9, Jet Propulsion Laboratory, Pasadena, California, September 1, 1981.

Table 1. The two-level model for the Viterbi decoder output statistics

	VLA-Goldstone array		VLA stand-alone system	
	Inside the gaps	Outside the gaps	Inside the gaps	Outside the gaps
Bit error rate	p_0	p_1	$p_0 \approx 1/2$	p_1
Symbol error rate	π_0	π_1	$\pi_0 \approx 1$	π_1
Signal-to-noise ratio	$\rho\, E_b/N_0$*	E_b/N_0	0	E_b/N_0

*Values of the array ratio considered in this article are: $10 \log_{10} \rho = 1.5$ dB, 3 dB, 5 dB (as well as $\rho = 0$ for the VLA stand-alone case).

Table 2. Code parameters, gap cycle parameters, and data rates

	General Case	Voyager-VLA Case
Convolutional code parameters		
Constraint length	K	7
Memory length	$K - 1$	6
Code rate	r	1/2
Reed-Solomon code parameters		
Symbol size	J	8
Codeword size	N	255
Error correction capability	E	16
Code rate	$1 - 2E/N$	223/255
Interleaving depth	I	4
Gap cycle parameters		
Gap length	G	1.6 msec
Total gap cycle length	T	5/96 sec
Data rate	R	21.6 kbits/sec
(Viterbi decoder bit rate)		14.4 kbits/sec
		7.2 kbits/sec
		3.6 kbits/sec

Table 3. Data rates causing major resonances between the gap cycle and the interleaving cycle

Resonant data rate (kbits/sec)	Worst-case number of gapped symbols per codeword
4.3008	37
4.9152	32
5.5296	29
6.1440	26
6.7584	24
7.3728	22
7.9872	20
8.6016	19
9.2160	17
9.8304	16
10.4448	15
11.0592	15
11.6736	14
12.2880	13
12.9024	13
13.5168	12
14.1312	12
14.7456	11
15.3600	11
15.9744	10
16.5888	10
17.2032	10
17.8176	9
18.4320	9
19.0464	18
19.6608	16
20.2752	16
20.8896	16
21.5040	16
22.1184	16

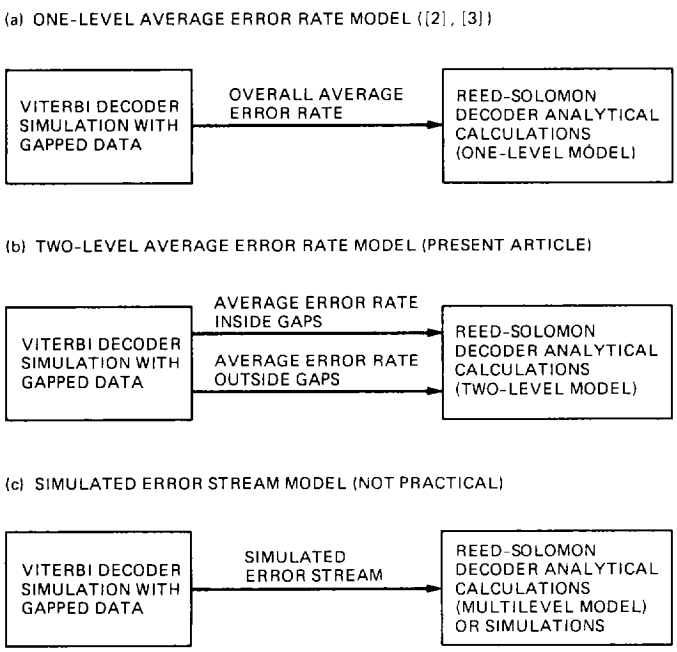

Fig. 1. Various analytical approaches to modeling concatenated coding with gapped data.

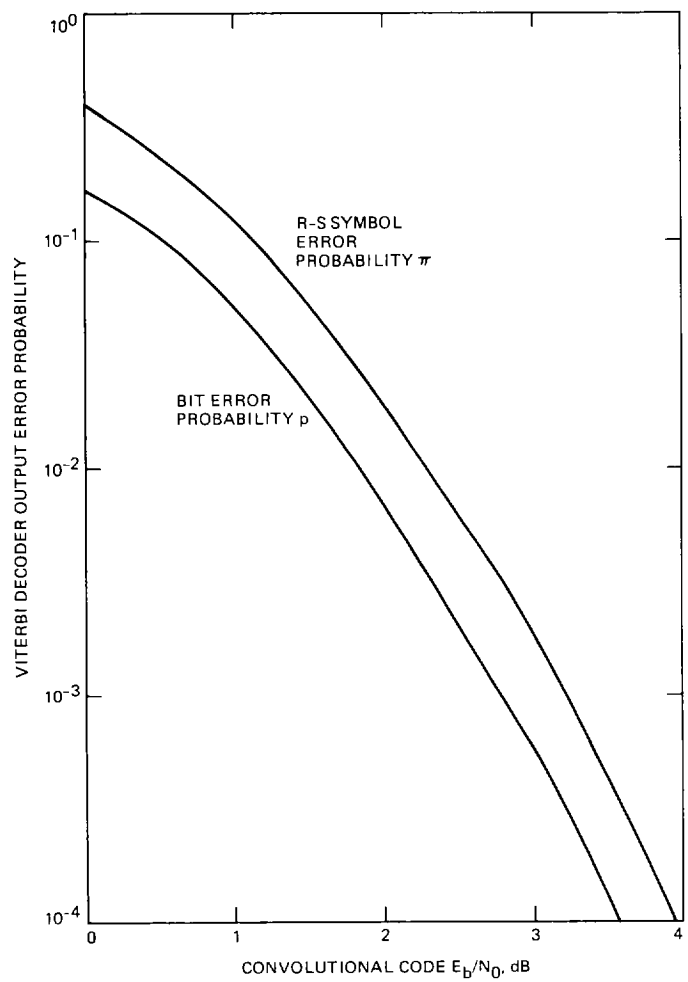

Fig. 2. Viterbi decoder output error probabilities for ungapped system (taken from [5], Fig. 3-1).

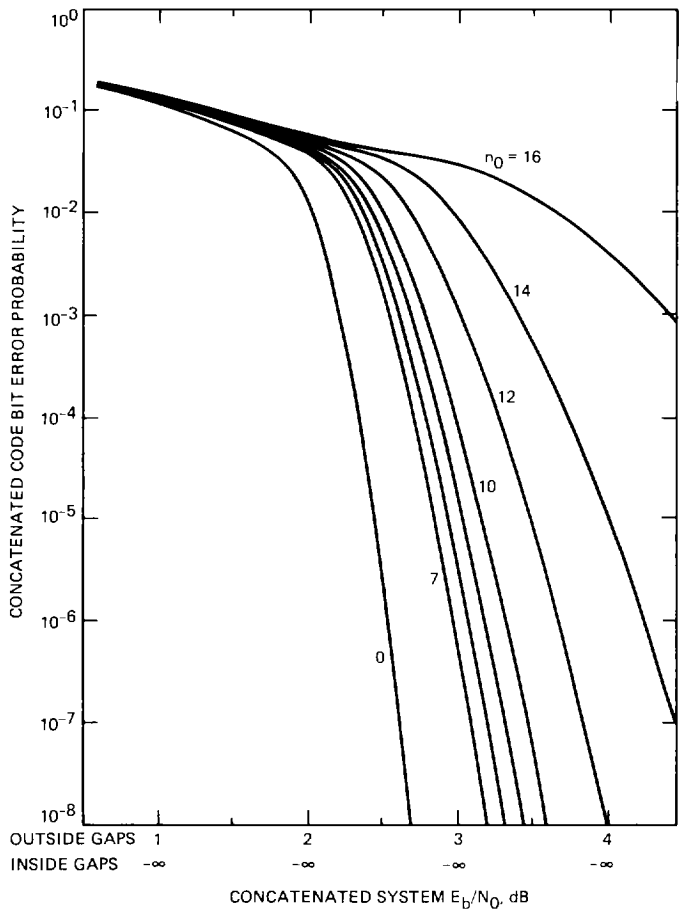

Fig. 3. Conditional concatenated performance for VLA stand-alone system as a function of n_0 = number of gapped symbols per codeword.

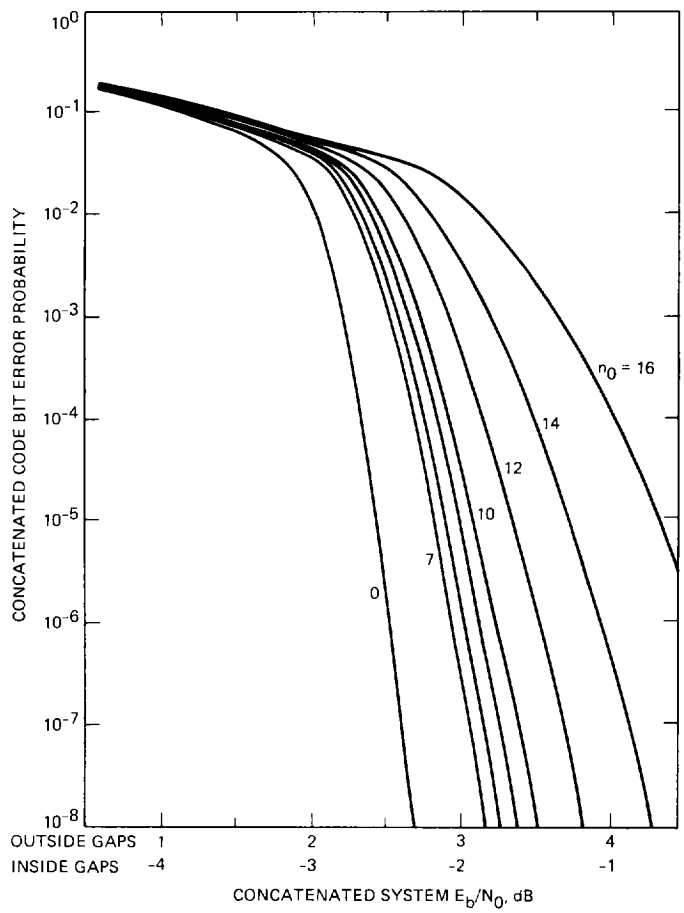

Fig. 4. Conditional concatenated code performance for VLA arrayed unequally with Goldstone (5-dB gaps) as a function of n_0 = number of gapped symbols per codeword.

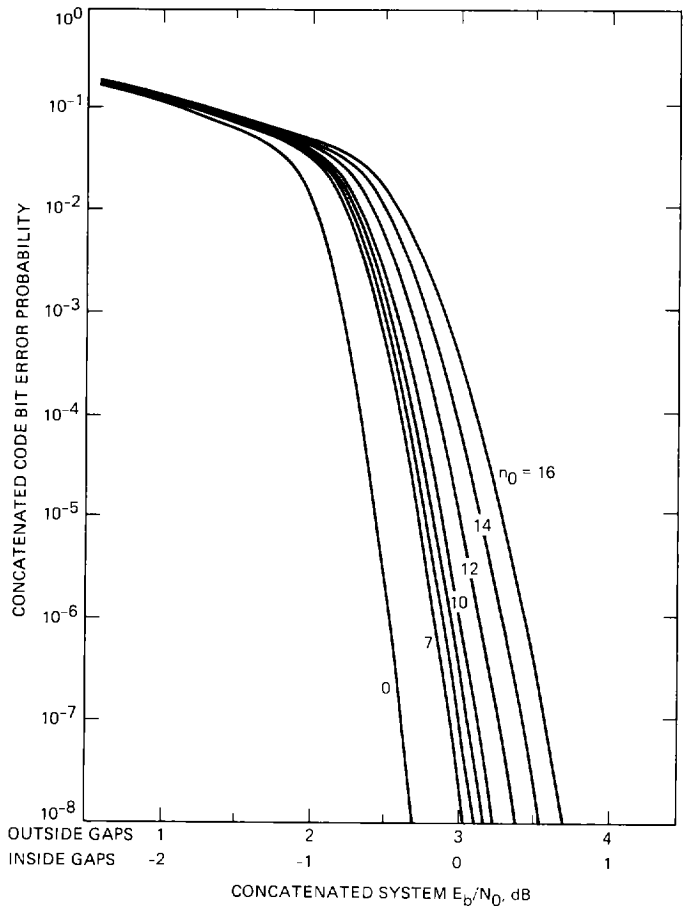

Fig. 5. Conditional concatenated code performance for VLA arrayed equally with Goldstone (3-dB gaps) as a function of n_0 = number of gapped symbols per codeword.

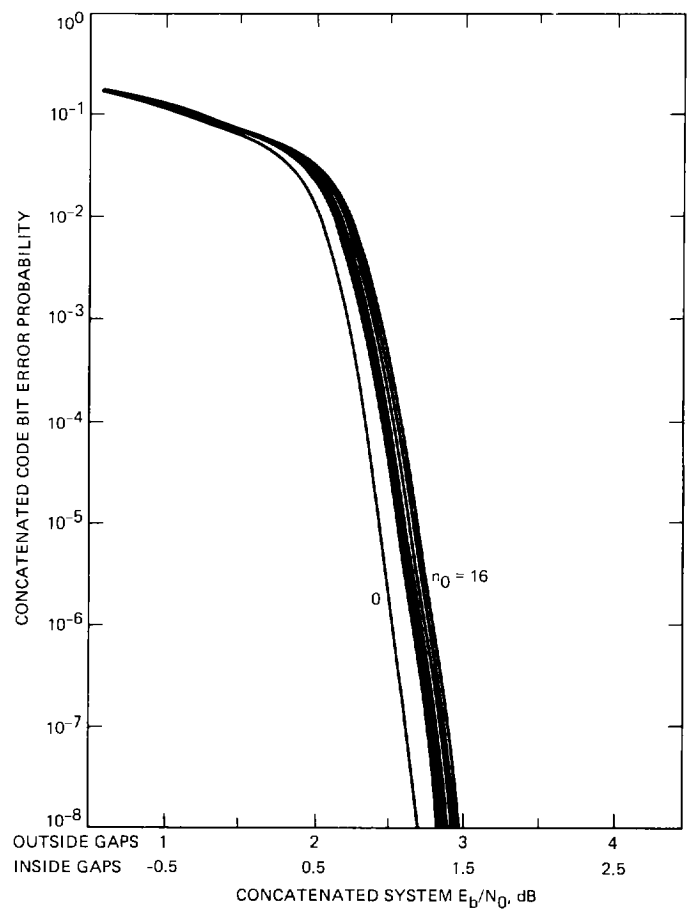

Fig. 6. Conditional concatenated code performance for VLA arrayed unequally with Goldstone (1.5-dB gaps) as a function of n_0 = number of gapped symbols per codeword.

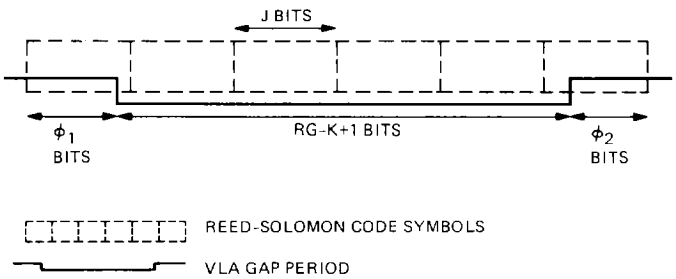

Fig. 7. Symbol edge effects.

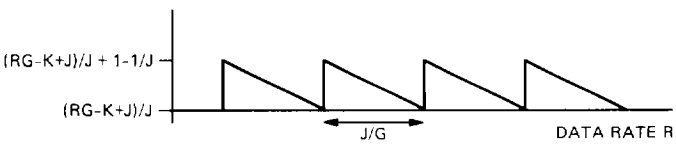

Fig. 8. Fluctuations in the number of gapped symbols per gap due to worst-case symbol edge effects.

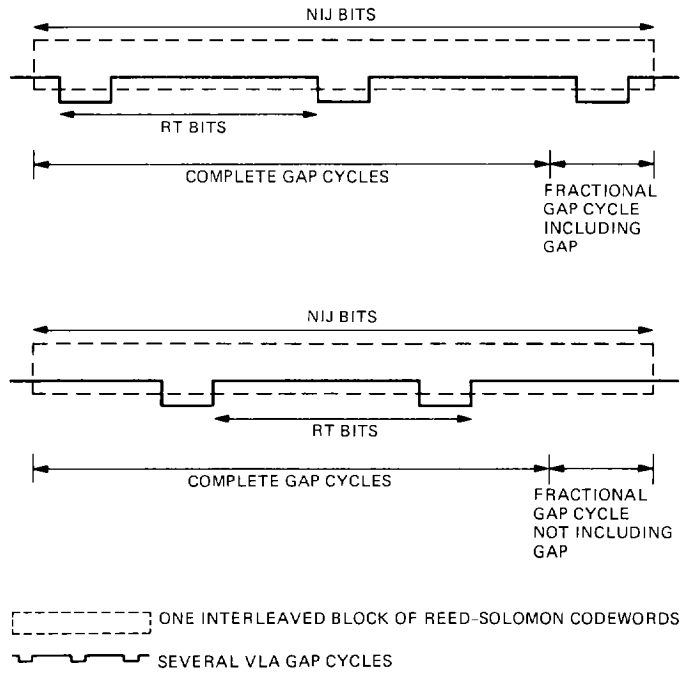

Fig. 9. Incomplete gap cycle effects.

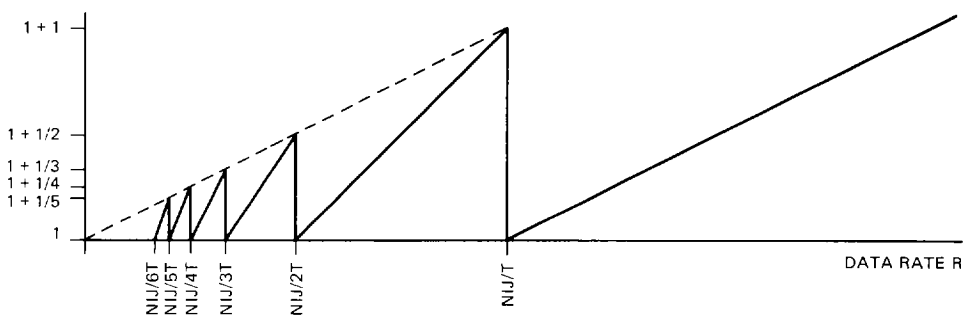

Fig. 10. Fluctuations in the peak-to-average concentration of gapped symbols per codeword due to incomplete gap cycles per interleaved block of codewords.

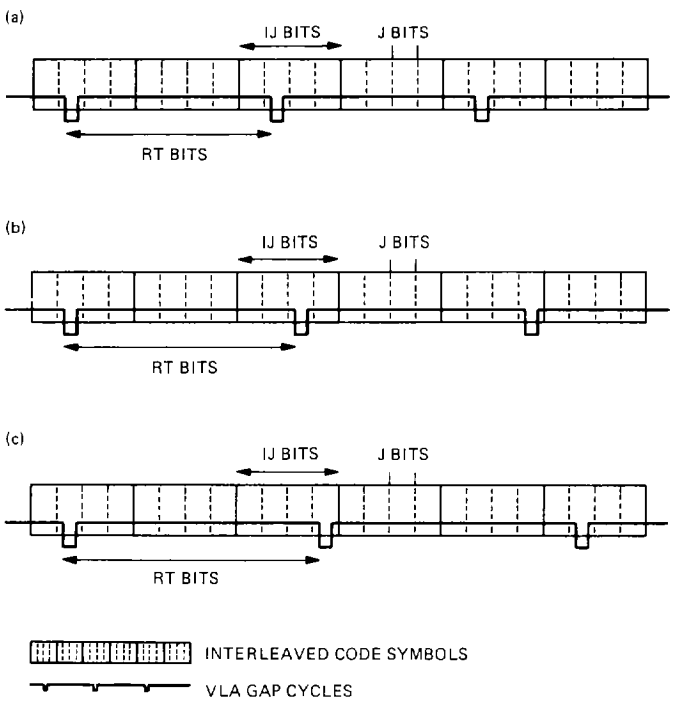

Fig. 11. Gap-cycle/interleaving-cycle resonance effects for average effective gap lengths up to one symbol.

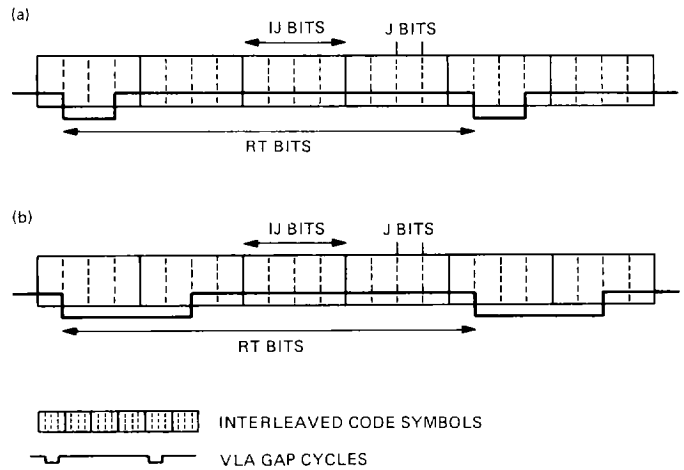

Fig. 12. Gap-cycle/interleaving-cycle resonance effects for average effective gap lengths greater than one symbol.

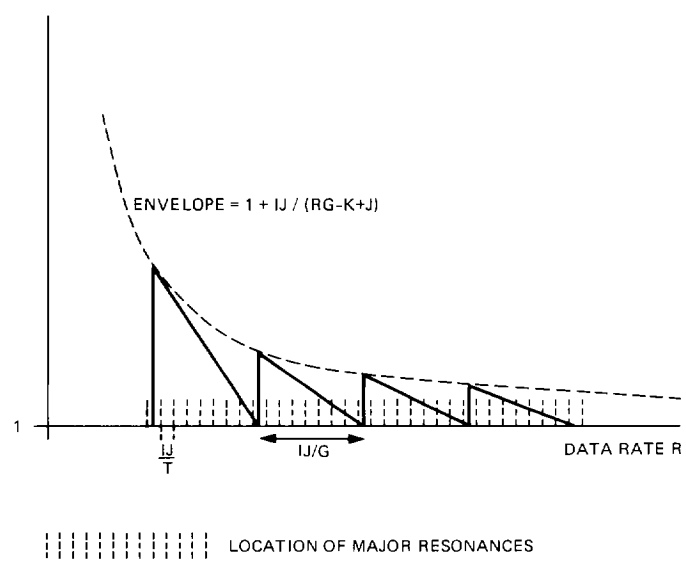

Fig. 13. Fluctuations in the peak-to-average concentration of gapped symbols per codeword due to gap-cycle/interleaving-cycle resonances. For clarity, figure depicts resonance spacing corresponding to T/G = 8. Actual T/G = 32.55 for the VLA parameters.

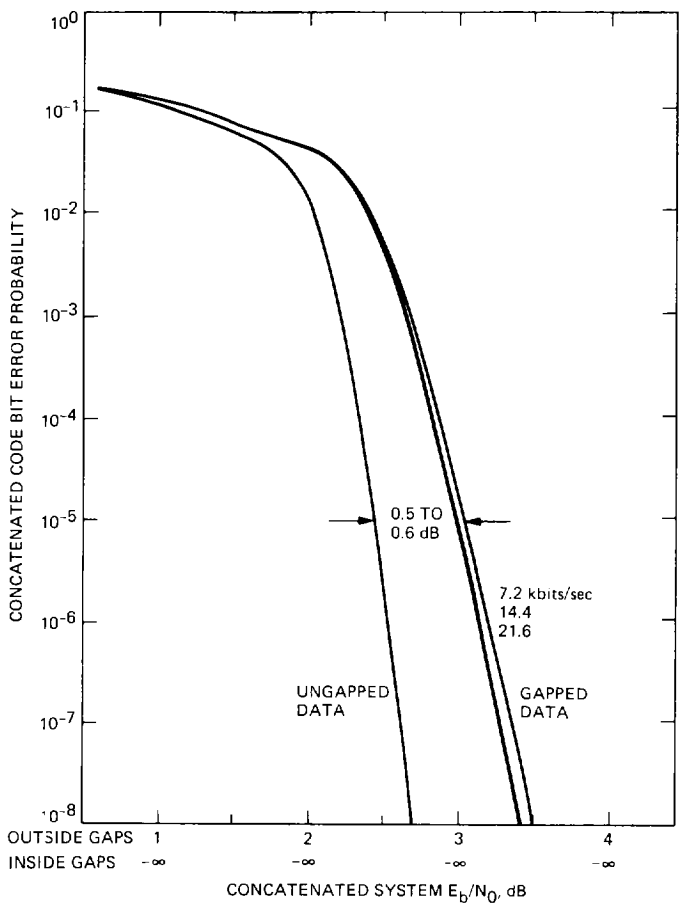

Fig. 14. Unconditional concatenated code performance at Voyager-Neptune data rates for VLA stand-alone system.

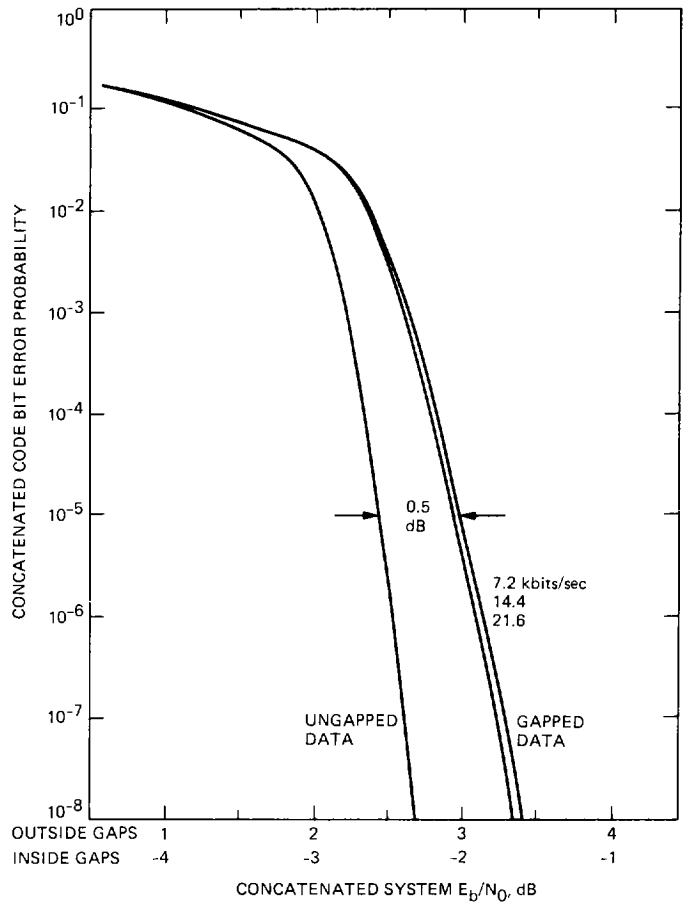

Fig. 15. Unconditional concatenated code performance at Voyager-Neptune data rates for VLA arrayed unequally with Goldstone (5-dB gaps).

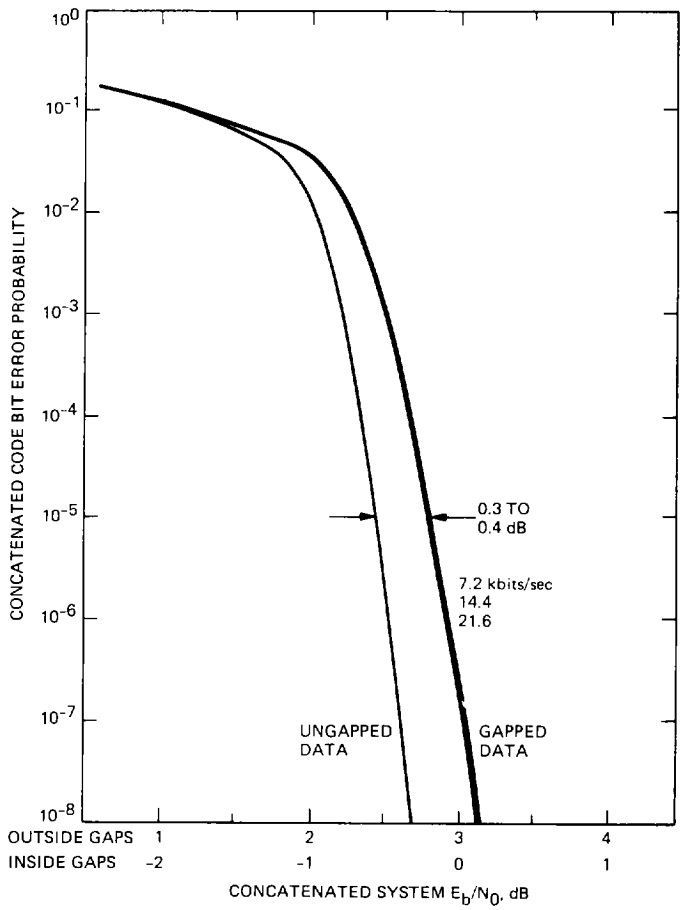

Fig. 16. Unconditional concatenated code performance at Voyager-Neptune data rates for VLA arrayed equally with Goldstone (3-dB gaps).

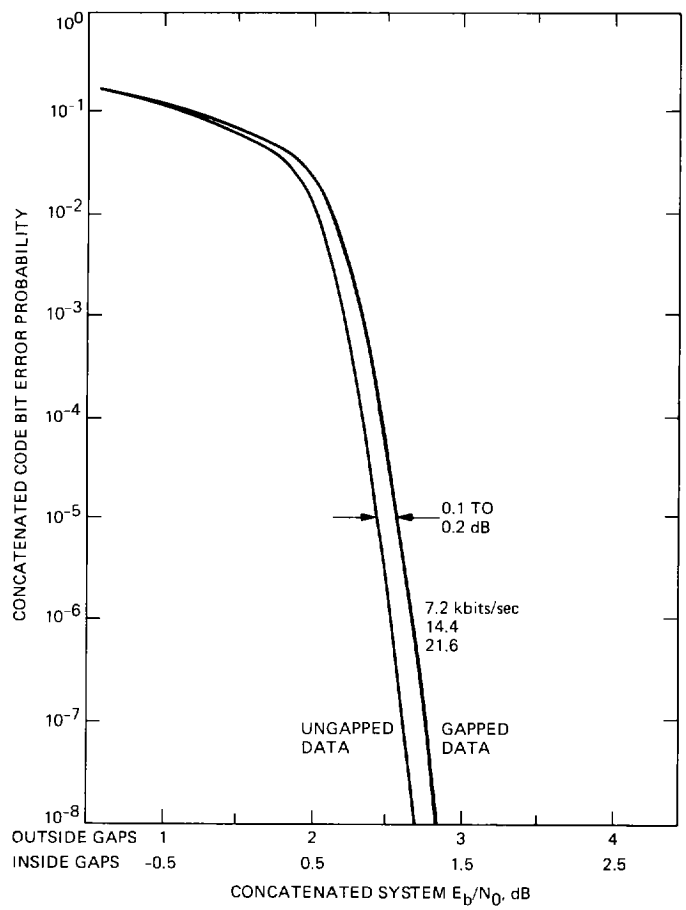

Fig. 17. Unconditional concatenated code performance at Voyager-Neptune data rates for VLA arrayed unequally with Goldstone (1.5-dB gaps).

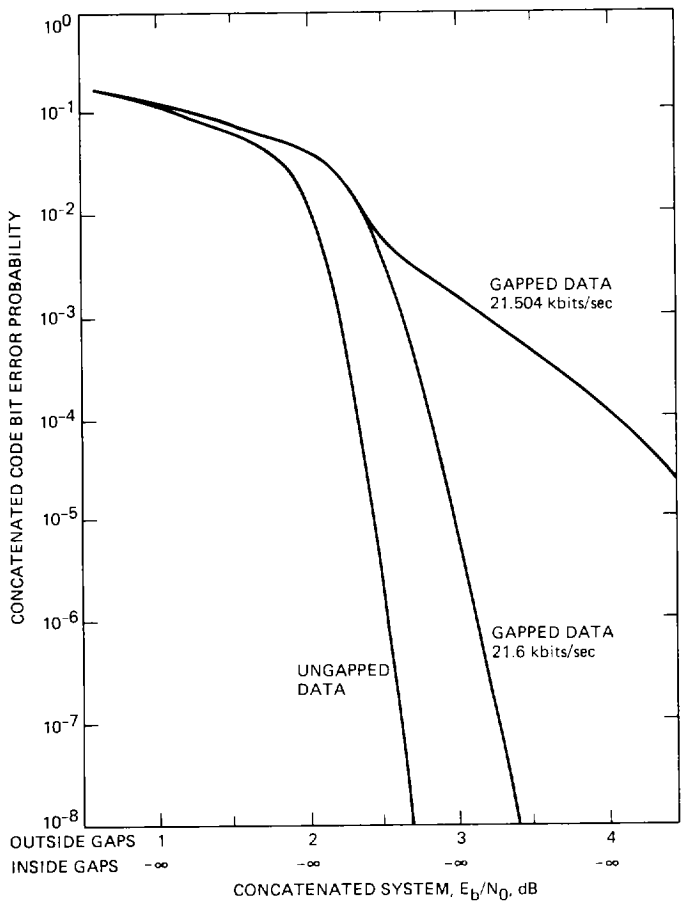

Fig. 18. Comparison of unconditional concatenated code performance at a Voyager-Neptune data rate and a nearby resonant data rate for VLA stand-alone system.

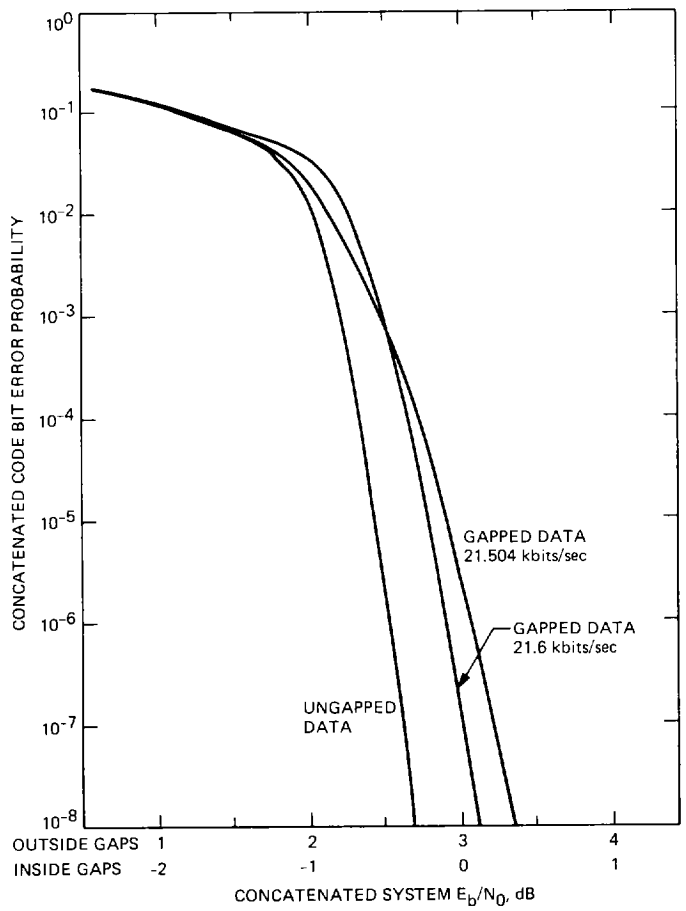

Fig. 19. Comparison of unconditional concatenated code performance at a Voyager-Neptune data rate and a nearby resonant data rate for VLA arrayed equally with Goldstone (3-dB gaps).

Appendix
Corroboration of the Two-Level Model

The two-level model for the Viterbi decoder output statistics is an ad hoc model that was chosen for simplicity. It is relatively easy to analyze, and it affords easy separability of the analysis into a conditional evaluation of code performance and a computation of the gap cycle's effect on the symbols in any given codeword. At the same time, it allows the gapped and ungapped portions of the gap cycle to be treated differently, not just characterized by overall cycle averages as in the even simpler one-level model.

The validity of the two-level model for the purposes of calculating concatenated code performance could be completely confirmed only by end-to-end tests or simulations of the entire concatenated code. Such simulations are impractical and were not performed. However, a partial corroboration of the model's accuracy can be obtained by comparing the average of the gapped and ungapped Viterbi decoder error rates predicted by the two-level models to the overall average Viterbi decoder error rates obtained from simulations of the Viterbi decoder operating over many VLA gap cycles. If the two-level model is accurate, the following equation should hold:

$$p = \frac{RG - K + 1}{RT} p_0 + \left(1 - \frac{RG - K + 1}{RT}\right) p_1 \qquad \text{(A-1)}$$

Here, p_0 and p_1 are the gapped and ungapped bit error probabilities defined in Table 1 and Fig. 2 and used in Eqs. (4) and (5), and p is the simulated bit error rate (averaged over many gap cycles) used in Eq. (1) of this article and plotted in Figs. 3 and 4 of [2].

Figures A-1 and A-2 compare the left sides and right sides of the purported Eq. (A-1) for the VLA stand-alone case and for the VLA arrayed equally with Goldstone.[1] Fairly good agreement is obtained. Even the substantial variation of performance with data rate is somewhat accurately predicted by the two-level model. The agreement is best for the VLA stand-alone case. The gap conditions in this case are more nearly equal to the assumption of zero signal-to-noise ratio inside the gaps and infinite signal-to-noise ratio outside the gaps, from which the argument for effectively shortening the gaps by $K - 1$ bits was derived.

A second form of justification for the two-level model is obtained by examining its validity at the end points. It was already stated in Section IV that the model is exactly correct in the extreme limit of zero signal-to-noise ratio inside the gaps and infinite signal-to-noise ratio outside the gaps. These extreme limits represent the maximum possible difference between the characteristics of the gapped and ungapped portions of the gap cycle. The opposite extreme occurs as the difference between the gapped zone and the ungapped zone goes to zero and the two signal-to-noise ratios become equal. The two-level model is obviously exactly correct in this extreme limit also, because Eqs. (2) through (7) reduce to Eq. (1) when $p_0 = p_1 = p$ and $\pi_0 = \pi_1 = \pi$. These end-point arguments do not directly confirm the validity of the two-level model in the intermediate regions, but nonetheless they inspire confidence that it should not be too far wrong.

[1]For comparison with [2], VLA gaps are assumed to be 1 msec instead of 1.6 msec for the curves in Figs. A-1 and A-2.

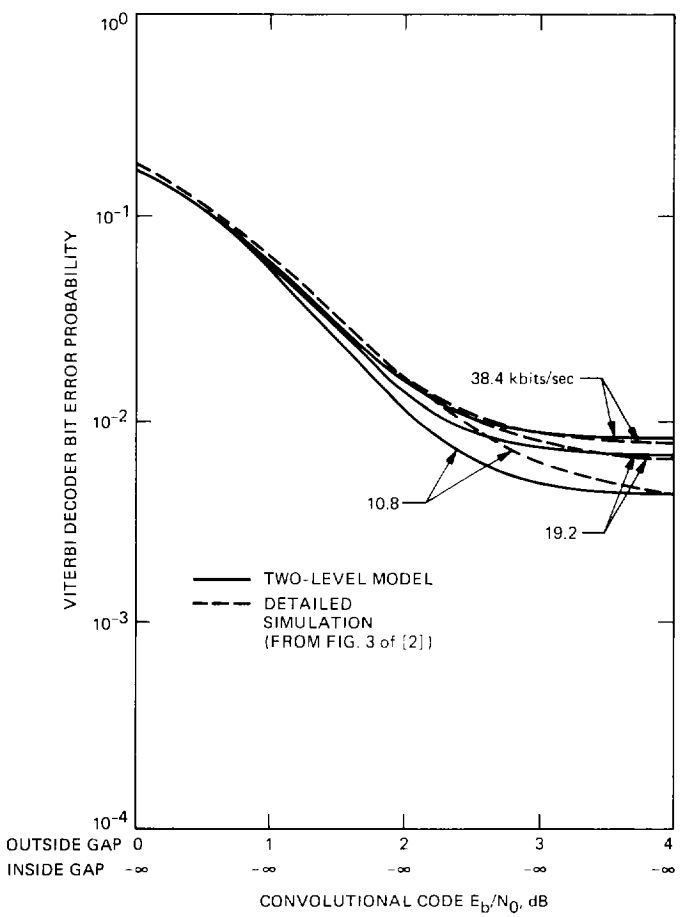

Fig. A-1. Comparison of Viterbi decoder performance predicted by two-level model and by detailed simulation for VLA stand-alone system. (Note: For comparison with [2], VLA gaps are assumed to be 1 msec instead of 1.6 msec for these curves.)

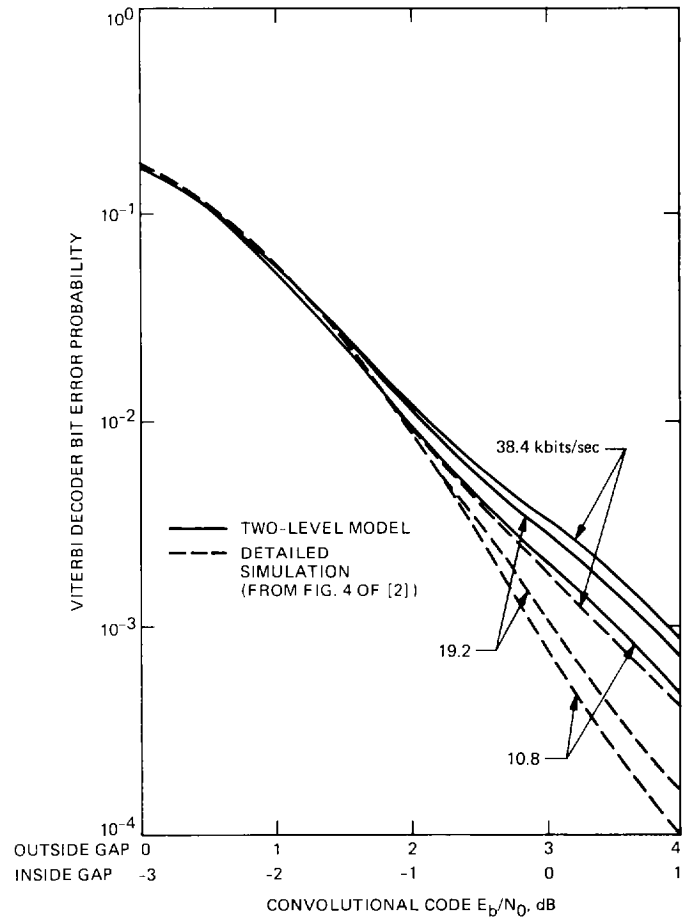

Fig. A-2. Comparison of Viterbi decoder performance predicted by two-level model and by detailed simulation for VLA arrayed equally with Goldstone (3-dB gaps). (Note: For comparison with [2], VLA gaps are assumed to be 1 msec instead of 1.6 msec for these curves.)