View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by NASA Technical Reports Server

N90-21896

Validity of the Two-Level Model for Viterbi Decoder Gap-Cycle Performance

S. Dolinar and S. Arnold
Communications Systems Research Section

A two-level model has previously been proposed for approximating the performance of a Viterbi decoder which encounters data received with periodically varying signal-to-noise ratio. Such cyclically gapped data is obtained from the Very Large Array (VLA), either operating as a stand-alone system or arrayed with Goldstone. This approximate model predicts that the decoder error rate will vary periodically between two discrete levels with the same period as the gap cycle. It further predicts that the length of the gapped portion of the decoder error cycle for a constraint length K decoder will be about K − 1 bits shorter than the actual duration of the gap.

In this article the two-level model for Viterbi decoder performance with gapped data is subjected to detailed validation tests. Curves showing the cyclical behavior of the decoder error burst statistics are compared with the simple square-wave cycles predicted by the model.

The validity of the model depends on a parameter often considered irrelevant in the analysis of Viterbi decoder performance, the overall scaling of the received signal or the decoder's branch-metrics. This article examines three scaling alternatives: optimum branch-metric scaling and constant branch-metric scaling combined with either constant noise-level scaling or constant signal-level scaling.

The simulated decoder error cycle curves roughly verify the accuracy of the two-level model for both the case of optimum branch-metric scaling and the case of constant branch-metric scaling combined with constant noise-level scaling. However, the model is not accurate for the case of constant branch-metric scaling combined with constant signal-level scaling.

I. Introduction

Voyager's concatenated code performance using gapped data as obtained from the Very Large Array (VLA) arrayed with Goldstone or the VLA operating as a stand-alone antenna was previously analyzed in [1]. This analysis rested on an assumed ad hoc approximate model for the errors produced by a Viterbi decoder when the signal-to-noise ratio (SNR) is subjected to variations between two discrete levels according to a fixed duty cycle. The model

of [1] predicted that the decoder error rate would also vary periodically between two discrete levels with the same period as the gap cycle. It further predicted that the duration of the gapped portion of the decoder error cycle for a constraint length K decoder would be about $K-1$ bits shorter than the actual VLA data gap.

The two-level model in [1] was generalized from an argument presented in [2] applicable to an ideal case in which the SNR inside the gap is zero ($-\infty$ dB), while the SNR outside the gap is infinite ($+\infty$ dB). The arguments of [2] predicted that the two-level model would be precisely accurate in this extreme case.

Simulated performance curves for gapped data were presented in [2] and in a follow-up article [3]. The curves in [2] were compared with predictions of the generalized two-level model in the appendix of [1]. Results showed fairly good agreement between simulations and model predictions, but some small discrepancies were also noted.

The performance curves of [2] and [3] and the validity checks in the appendix of [1] were limited in that they only considered overall average Viterbi decoder error rates. This article looks at the decoder's error characteristics in finer detail by examining the cyclical variations of error rate over a gap cycle. This level of detail is necessary for a full direct validation of the model, rather than just an incomplete indirect confirmation of the model based on average error rates.

In the course of the new investigation, it was discovered that a parameter usually considered irrelevant in the analysis of Viterbi decoder performance, the overall scaling of the received signal or the decoder's branch metrics, must be carefully accounted for in order to ensure maximum-likelihood performance. When the received-signal characteristics (e.g., SNR) change abruptly, so should the scaling of branch metrics. If the signal characteristics fluctuate, a Viterbi decoder designed with constant branch metric scaling is not a maximum-likelihood decoder unless the received signal is rescaled optimally. Since hardware and software versions of the Viterbi decoder often have constant branch-metric scaling built in, the effects of nonoptimum received signal scaling on these decoders must be evaluated.

Fortunately, the scaling issue seems moot for most of the VLA applications and analysis performed up to now. The optimum weights for combining the VLA and Goldstone received signals happen to provide the exact scaling required for optimum performance by a decoder with constant branch-metric scaling. The simulations of [2] and [3] appear to have been based on near-optimum scaling. However, most of the Viterbi-decoder simulation software developed recently for nongapped scenarios is based on nonoptimum received-signal scaling, which can greatly widen the effective gap if used by a decoder with constant branch-metric scaling. Therefore, extreme care must be exercised in adapting existing software to gapped scenarios.

II. Method of Analysis

A. Decoder Error Cycles

When the SNR varies over a gap cycle, the errors made by the decoder can be expected to vary cyclically with the same period. In order to determine the variation of decoder performance over a gap cycle, it is necessary to obtain some estimate of the decoder's instantaneous error rate at every point within the cycle, rather than to simple average its error rate over the entire simulation run. Knowledge of the decoder's error-cycle characteristics is especially important when concatenated coding is used and the convolutionally decoded output must be further decoded according to an outer code.

As a measure of decoder performance, we chose to calculate the *burst inclusion probability* (BIP) rather than the bit error rate (BER) or the symbol error rate (SER). A given bit is included within an error burst if it is not part of a string of $K-1$ or more consecutive correctly decoded bits, where K is the constraint length of the convolutional code. This definition is equivalent to the definition of an error burst given in [4]. The burst inclusion probability at any particular point in the gap cycle is the probability that a bit at that location relative to the gap will be included within a decoder error burst. The BIP can be used to give reasonable estimates of either the BER or the SER for various symbol sizes, and the cyclical variation of the BIP accurately reflects the cyclical variation of the decoder's instantaneous BER or SER.

If the SNR is constant over a gap cycle, the gap is effectively missing and the BIP is also a constant over the entire cycle. However, when the SNR switches between two levels, SNR_0 inside the gap and SNR_1 outside the gap, the BIP will also oscillate. The model of [1] predicts the magnitudes of the peaks and troughs of this oscillation and the width of the peaks. It also predicts instantaneous jumps from peak to trough and vice versa. The simulations reported here investigate the accuracy of these predictions. How big are the actual peaks and troughs? How wide are the actual peaks? What are the actual rise times and fall times?

The two-level model of [1] predicts simple square-wave error cycles. The BIP should oscillate between two levels, BIP_0 and BIP_1, for "gapped" and "ungapped" bits, respectively. A discrete jump from level BIP_1 to level BIP_0 should occur at the beginning of every gap, and a discrete jump from level BIP_0 to level BIP_1 should occur $K-1$ bits earlier than the end of every gap. Furthermore, the model predicts that these two discrete levels, BIP_0 and BIP_1, are the same as the constant BIP levels that would prevail at constant SNR levels SNR_0 and SNR_1, respectively. The model also predicts the same square-wave cycles for the decoder's BER and SER, but these predictions are only tested indirectly in this article by measuring the burst inclusion probability.

B. Description of Simulations Performed

Simulations were run for a Viterbi decoder operating on data encoded by the NASA-standard (7,1/2) code. The simulated decoder generated its own random noise samples, based its decisions on unquantized metrics, and used a decoding truncation window of 64 bits.

The decoder was confronted with a periodically varying SNR. The SNR varied between two levels in synchronism with a gap cycle of fixed period T. For a data rate R, the gap cycle measured in decoded bits is of length RT. Within every gap cycle is a gap of length G, or RG bits, during which the SNR is at its lower level, $SNR = SNR_0$. For the remainder of each cycle, the SNR is at its higher level, $SNR = SNR_1$.

In the simulations reported here, the full gap cycle was taken to be $RT = 1123$ bits, and the gapped portion of the cycle was $RG = 35$ bits. This corresponds to the VLA gap cycle parameters used in [3] for the highest Voyager-Neptune data rate of $R = 21.6$ kbps.

Each simulation decoded a total of 800,000 bits, just over 712 full gap cycles. Separate simulations were run for many different combinations of SNR_0 and SNR_1.

III. Viterbi Decoder Scaling Considerations

Viterbi decoders are supposed to operate as maximum-likelihood decoders. However, real-world Viterbi decoders, such as the DSN's maximum-likelihood convolutional decoder (MCD), may be designed on the assumption of constant SNR or slowly varying SNR. Similarly, software simulations of Viterbi decoder operations usually implicitly assume nonvarying SNR. Such decoders and simulations of decoders are significantly suboptimum if the received-signal characteristics change abruptly within a time period comparable to a few constraint lengths.

There are two types of scaling considerations that become important when the SNR is time-varying. How is the received signal scaled? How are the decoder's branch metrics scaled? With nonvarying SNR, the overall scaling of the received signal and the branch metrics is irrelevant. They may all be multiplied or divided by any convenient scale factor and the decoder will produce the same results. When the SNR is time-varying, the scale factor does matter. To be optimum, the scaling of the branch metrics and/or the received signal must change as the signal and noise characteristics vary with time.

A. Optimum Branch-Metric Scaling

The model for the received symbol r_k is

$$r_k = m_k d_k + \sigma_k n_k \tag{1}$$

where $d_k = \pm 1$ is the kth encoded symbol, n_k is zero-mean, unit-variance Gaussian noise, and m_k, σ_k are the signal level and noise standard deviation, respectively, for the kth received symbol. The conditional probability density function for a set of received symbols $\{r_k\}$ computed along a path corresponding to encoded symbols $\{c_{ik}\}$ for the ith codeword ($c_{ik} = \pm 1$) is

$$p_i(\{r_k\}) = \prod_k \frac{1}{\sqrt{2\pi\sigma_k^2}} e^{-(r_k - m_k c_{ik})^2/2\sigma_k^2} \tag{2}$$

A maximum-likelihood metric[1] M_i can be obtained from Eq. (2) by taking logarithms and ignoring some terms which are independent of i.

$$M_i = \sum_k \frac{1}{2}\left[|r_k| - c_{ik} r_k\right] \alpha_k \tag{3}$$

where α_k is the optimum branch-metric scaling factor,

$$\alpha_k = \frac{m_k}{\sigma_k^2} \tag{4}$$

When the signal characteristics are constant with time, the scale factor α_k is also constant and may be disregarded, because an overall scaling of all metrics does not

[1] The metric in Eq. (3) is called the "sign-magnitude metric" because $\frac{1}{2}\left[|r_k| - c_{ik} r_k\right]$ equals $|r_k|$ if c_{ik} and r_k have opposite signs, or 0 if they have the same sign. This metric is obtained by subtracting the usual correlation metric from $\sum_k |r_k|\alpha_k$ and dividing by 2.

matter. Both hardware decoders and simulated decoders often have this assumption built into their computation of branch metrics, i.e., they assume $\alpha_k = 1$ (or some other suitable constant).

B. Received-Signal Scaling

If the signal characteristics vary with time, scaling may also be performed on the received signal itself. Some sort of automatic gain control may be functioning to keep the received signal within the dynamic range of the receiver. When baseband combining is performed, additional weights are applied to each separate component of the received signal. If these scale factors change quickly enough during a gap cycle, the resulting variation in α_k must be accommodated by the Viterbi decoder, or else the decoder is not maximum likelihood.

If the decoder cannot adjust the optimum branch-metric weights α_k dynamically, optimum performance may still be achieved if the received signal is scaled in such a way that $\alpha_k = 1$ for all k. This optimum scaling of the received signal is

$$r_k = \rho_k d_k + \sqrt{\rho_k} n_k \qquad \text{(optimum scaling)} \qquad (5)$$

where ρ_k is twice the instantaneous channel symbol SNR,

$$\rho_k = \frac{m_k^2}{\sigma_k^2} = \frac{2}{N}\left(\frac{E_b}{N_0}\right)_k \qquad (6)$$

In Eq. (6), $1/N$ is the rate of the convolutional code, and $(E_b/N_0)_k$ is the bit-energy-to-noise ratio for the kth symbol.

Two other nonoptimum received-signal scalings have usually been assumed in software simulations of the Viterbi decoder. Sometimes the signal level is assumed to be a constant (± 1), and sometimes the standard deviation of the noise is assumed to be constant. These two standard scaling assumptions can be written compactly in terms of the parameter ρ_k as

$$r_k = d_k + \frac{n_k}{\sqrt{\rho_k}} \qquad \text{(constant signal-level scaling)} \qquad (7)$$

or

$$r_k = \sqrt{\rho_k} d_k + n_k \qquad \text{(constant noise-level scaling)} \qquad (8)$$

C. Scaling Used in Existing Software Simulations and DSN Hardware

Historically, most versions of the Communications Systems Research Section's Viterbi decoder software have assumed constant branch-metric scaling. The DSN's hardware MCD for the NASA-standard (7,1/2) code also makes use of constant branch-metric scaling. The Big Viterbi Decoder (BVD) currently under development has programmable branch-metric scaling.[2]

Most recent simulation software has incorporated constant signal-level scaling of the received symbols. Some older simulations, apparently[3] including the VLA gapped data simulations reported in [2] and [3], were based on constant noise-level scaling.

The DSN's current hardware uses an automatic gain control circuit (AGC), but the response time is far too slow[4] to cause any fluctuation during a VLA gap. In a VLA-Goldstone arrayed system, optimum combining weights must be applied in order to fully utilize the SNR of each array component. It can be shown from Eq. (20) of [5] that the scaling produced by applying the optimum weights for baseband combining is equivalent to the optimum scaling for a decoder using constant branch-metric scaling, provided that the signal characteristics at the reference antenna are unchanging. This should be true if Goldstone is the reference antenna. In a VLA stand-alone system, the scaling will be optimum if the received data is ignored during the gap, i.e., if all the branch metrics during the gap are set to zero.

D. Effects of Received-Signal Scaling on Decoders With Constant Branch-Metric Scaling

If the decoder is constrained to have constant branch-metric scaling, the decoder's error characteristics are very much influenced by the received-signal scaling. Some appreciation of these differences may be gained by reanalyzing the ideal gap scenario upon which the model of [1] is based, as exposited in [2]. If the SNR outside the gap is infinite and the SNR inside the gap is zero, the decoder should have perfect state information at all places outside the gap, including the two bit times positioned at the brink of the gap. Since the state information at the end of the gap includes $K-1$ bits that were transmitted during the gap, knowledge of the state at the end of the gap allows the decoder to correctly decode $K-1$ gapped bits in addition to all the bits outside the gap.

[2] Oliver Collins, personal communication.

[3] The scaling assumed in [2] and [3] was not documented, and this tentative conclusion can only be inferred by deciphering the original computer code, some of which was written in assembly language.

[4] Joseph Statman, personal communication.

However, this is not true if nonoptimum branch-metric scaling is used. For example, consider the case of *constant branch-metric scaling combined with constant signal-level scaling.* At the leading edge of the gap, the accumulated metrics will have all reached some distribution of steady-state values corresponding to finite signal and zero noise according to Eq. (7). At the very next bit time, if the SNR drops nearly to zero, the noise component of Eq. (7) will completely overwhelm the data component, and the branch metrics will all be random and huge. Large random differences in the branch metrics will wipe out any steady-state differences in accumulated state metrics that were built up over the entire ungapped period. Large random branch metrics will continue to be added throughout the gap, resulting in a state-metric distribution at the end of the gap which is completely random and very large. Whichever state metric is smallest at the end of the gap is likely to be smallest by a large margin over the second smallest metric. After the gap is finished, the noise component of Eq. (7) returns to zero, the branch metrics revert to their usual magnitudes and all point toward the correct state. However, the metric differences arising from adding these smaller branch metrics following the gap are unlikely to be large enough to overcome the huge accumulated metric difference in favor of the random state with the smallest metric at the end of the gap. Thus, the decoded state path will almost assuredly pass through this random state. The best-metric state path traced backward from this random state will find its way to another random state at the start of the gap.

Thus, with constant signal-level scaling, the decoded state path passes through a random state at the beginning of the gap and another random state at the end of the gap. The decoded bits are totally random from the beginning of the gap to the end. Also, there are additional decoded bit errors before the beginning of the gap along the best-metric path from the correct state to the first random state, and after the end of the gap along the best-metric path from the last random state back to the correct state. Thus, the gap is effectively widened by some number of bits on both ends.

Now consider the case of *constant branch-metric scaling combined with constant noise-level scaling of the received signal.* In this case, all the accumulated metrics at the beginning of the gap will have reached the same relative steady-state distribution as before, but all differences in accumulated metrics will be magnified by a scale factor $\sqrt{\rho}$ related to the high SNR outside the gap. The branch metrics during the gap will be just as random as before, but their magnitudes will be small relative to the accumulated metric differences at the beginning of the gap. At the end of the gap, a random state will have the best metric, but the metrics of all states will be small relative to the metric differences that are obtained along the path portions outside the gap. After the end of the gap, the decoder will eventually find the correct state and will trace the correct state path all the way back to the trailing edge of the gap. The branch-metric information accumulated after the gap outweighs the random state-metric differences existing at the end of the gap, and therefore the correct state path may be traced all the way to the gap edge.

Constant noise-level scaling thus corresponds to the ideal case considered originally, where the decoder knows the correct state both at the beginning and at the end of the gap. The random path followed between these two states during the gap corresponds to a string of random decoded bits which is $K-1$ bits shorter than the actual length of the gap. A similar argument reaches the same conclusion for the case of optimum scaling, or, equivalently, for arbitrary received-signal scaling combined with optimum branch-metric scaling.

In summary, both optimum scaling and constant noise-level scaling produce decoded error cycles that precisely match the predicted model in the extreme case of zero SNR inside the gap and infinite SNR outside the gap. In contrast, constant signal-level scaling does not fit the model even in this ideal case. The simulation results in the next section generalize these conclusions to nonideal cases having nonzero, finite SNR levels.

IV. Decoder Error-Cycle Curves

The burst inclusion probability (BIP) was evaluated as a function of location within a gap cycle for the three types of received-signal scaling defined above, assuming that the branch-metric computations used a constant scale factor $\alpha_k = 1$. As noted earlier, the case of "optimum" scaling of the received signal is equivalent to arbitrary signal scaling combined with optimum branch-metric scaling, so this case shows the performance of a true maximum-likelihood decoder. The BIP was evaluated for various combinations of SNR_0 and SNR_1. These curves were plotted against a background showing the effective gap width and the constant error levels predicted by the two-level model in [1].

Figures 1(a), (b), and (c) show the error-cycle curves for a series of cases with $SNR_0 = 0$ (the VLA stand-alone case considered in [1]). The SNR_1 values for the five curves plotted were 1 dB, 2 dB, 3 dB, 4 dB, and $+\infty$ dB. Figure 1(a) plots these five curves for optimum branch-metric

scaling, Fig. 1(b) plots the same curves assuming constant noise-level scaling, and Fig. 1(c) plots them for constant signal-level scaling.

Figures 2(a), (b), and (c) and 3(a), (b), and (c) show the error-cycle curves for a series of cases with SNR_0 3 dB lower than SNR_1 (the VLA-Goldstone equal array case considered in [1]). The SNR_1 values for the four curves plotted in Figs. 2(a), (b), and (c) are 1 dB, 2 dB, 3 dB, and 4 dB. The SNR_1 values for the three curves plotted in Figs. 3(a), (b), and (c) are 1.8 dB, 2 dB, and 2.2 dB.

Figures 4(a), (b), and (c) show the error-cycle curves for a series of cases with SNR_0 2 dB lower than SNR_1. The SNR_1 values for the four curves plotted in Figs. 4(a), (b), and (c) are 1 dB, 2 dB, 3 dB, and 4 dB.

In each figure, dashed horizontal lines show constant BIP levels corresponding to constant SNR levels indicated on the right vertical axis. These are the constant BIP levels predicted by the two-level model for the decoder error-cycle curves. Dashed vertical lines on each figure show the predicted effective beginning and end of the gap. The effective beginning equals the beginning of the actual gap, and the effective end is $K-1$ bits earlier than the actual end. If the two-level model is correct, the simulated decoder error-cycle curves should be framed exactly by a square wave constructed from the appropriate combination of dashed horizontal and vertical lines.

It is seen in Figs. 1(a) and (b) that the two-level model of [1] is precisely correct[5] (as expected) for the extreme case of zero SNR inside the gap and infinite SNR outside the gap. When the SNR outside the gap is not infinite, the burst inclusion probability experiences a vertical drop-off precisely at the points predicted by the model, but this sharp drop-off slackens before reaching the predicted error rate outside the gaps. Thus, the effective gap length is approximately $RG-K+1$ bits as predicted by the model, but it is widened slightly due to the "wings" that appear in the curves of Figs. 1(a) and (b). The model of [1] *slightly underestimates* the decoder's errors by squaring off these curves and eliminating these wings. This finding is consistent with the slight underestimate of decoder errors by the two-level model shown in Fig. A-1 of [1] for the VLA stand-alone case.

[5] The slight rounding of the error-cycle curves near the edges of the effective gap window occurs because, in our definition, bursts must start and end with actual error bits. Some random bits at the beginning or end of the effective gap which happen to be decoded correctly are therefore not classified as being included within a burst, even though they are random.

In contrast, the curves in Fig. 1(c) show that the two-level model of [1] *significantly underestimates* the effective width of the gaps, as predicted by the arguments at the end of the previous section. Not only is the region of maximum error rate longer than the effective gap length $RG-K+1$ by about 4 to 5 bits on each side, but the drop-off from the maximum error rate is not as sharp and the "wings" are much wider.

Figures 2(a), (b), 3(a), (b), and 4(a), (b) show less definitive results for the cases of nonzero SNR inside the gap. There are clearly wings that extend the effective length of the gaps but at greatly reduced error levels, as in Figs. 1(a) and (b). However, the burst inclusion probability inside the gap is also rounded off (as compared to the model's prediction) and does not rise vertically to the predicted steady-state level. Figure 5 is a replot of one of the curves of Fig. 2(a) on a linear scale. On this figure it can be clearly seen that the model *slightly overestimates* the decoder's total errors. The area under the curve saved by not quickly rising to the steady-state level inside the gap is greater than the area lost by not quickly falling to the steady-state level outside the gap. This finding offers an explanation for the curves in Fig. A-2 of [1], where it was seen that the two-level model predicted slightly worse performance than the simulation for the case of VLA-Goldstone equal array (3-dB gaps).

Figures 2(c), 3(c), and 4(c) show the same types of tradeoffs. The error-cycle curves do not fall as sharply as predicted to their steady-state levels outside the gap, nor do they rise as sharply as predicted to their steady-state levels inside the gap. However, the error-cycle peaks are clearly several bits wider than the corresponding ones in Figs. 2(a), (b), 3(a), (b), or 4(a), (b).

V. Conclusions

In this study the two-level model for Viterbi decoder performance with gapped data was subjected to more-detailed validation tests than were performed in [1]. Detailed curves showing the cyclical behavior of the decoder error burst statistics were compared with the simple square-wave cycles predicted by the model.

The simulated decoder error-cycle curves roughly verify the accuracy of the two-level model for both the case of optimum branch-metric scaling and the case of constant branch-metric scaling combined with constant noise-level scaling of the received signal. Minor discrepancies between the model and the simulation include a broadening of the effective gap by several bits on both sides, but at a greatly reduced error level, whenever the SNR outside the gap is

not infinite, and a lowering of the gap error rate, especially near the edges, whenever the SNR inside the gap is not zero. These compensating effects lead to a slight net underestimate of the overall error rate by the model in the case of a VLA stand-alone system, and a slight net overestimate of the error rate in the case of the VLA arrayed equally with Goldstone. Both of these effects are small and do not critically diminish the validity of the model.

The model is not accurate for the case of constant branch-metric scaling combined with constant signal-level scaling. The effective gap in this case is significantly wider than that predicted by the model, and the error rate falls much less rapidly than predicted outside the gap. Therefore, it is important to avoid this combination of received-signal scaling and branch-metric scaling both in hardware systems and in software simulations.

References

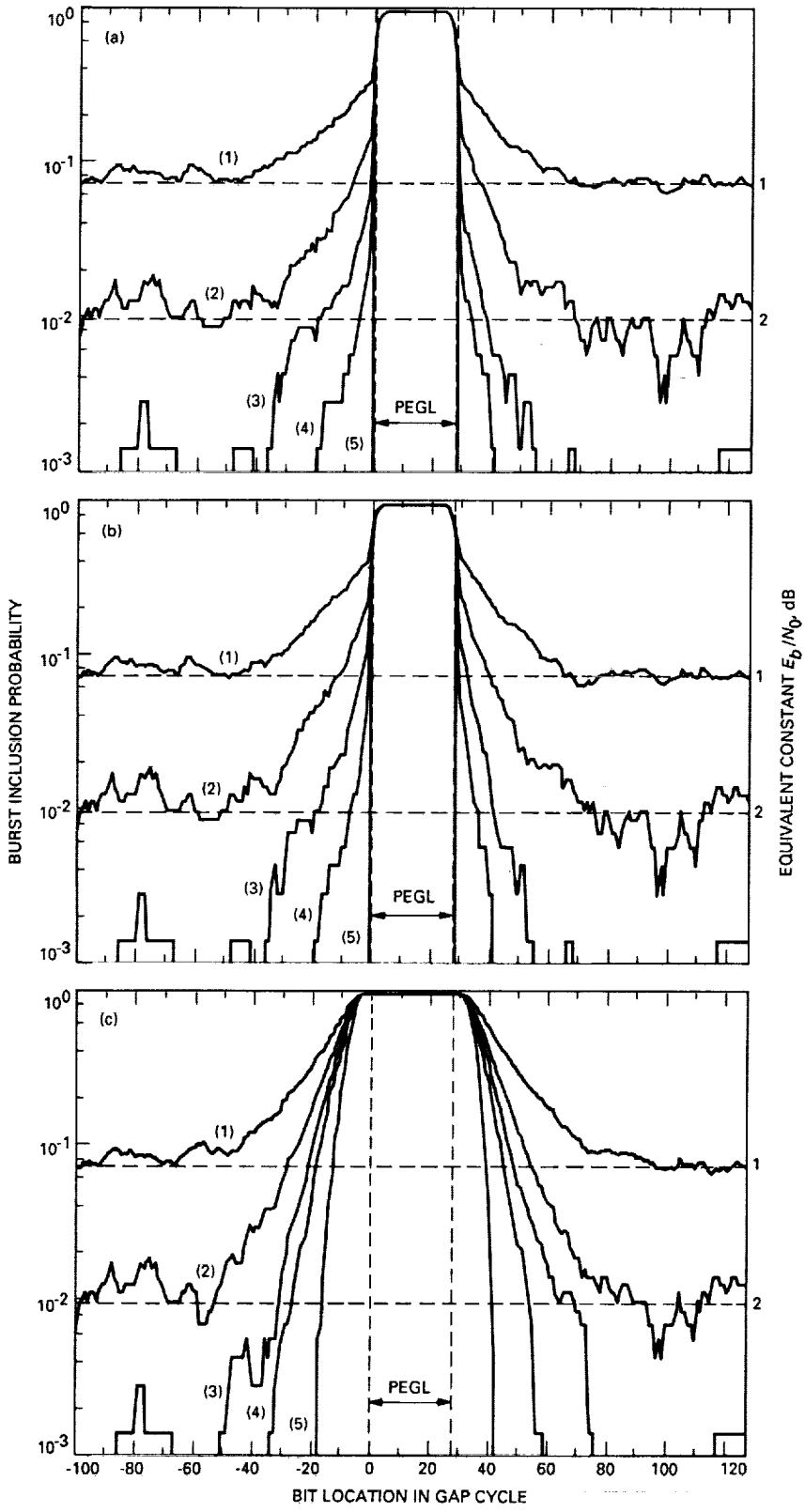
[1] S. J. Dolinar, "VLA Telemetry Performance With Concatenated Coding for Voyager at Neptune," *TDA Progress Report 42-95,* vol. July–September 1988, Jet Propulsion Laboratory, Pasadena, California, pp. 112–133, November 15, 1988.

[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, Jet Propulsion Laboratory, Pasadena, California, pp. 27–39, November 15, 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, Jet Propulsion Laboratory, Pasadena, California, pp. 51–60, 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.

[5] D. Divsalar, D. Hansen, and J. H. Yuen, "The Effect of Noisy Carrier Reference on Telemetry with Baseband Arraying," *TDA Progress Report 42-63,* vol. March and April 1981, Jet Propulsion Laboratory, Pasadena, California, pp. 128–135, June 15, 1981.

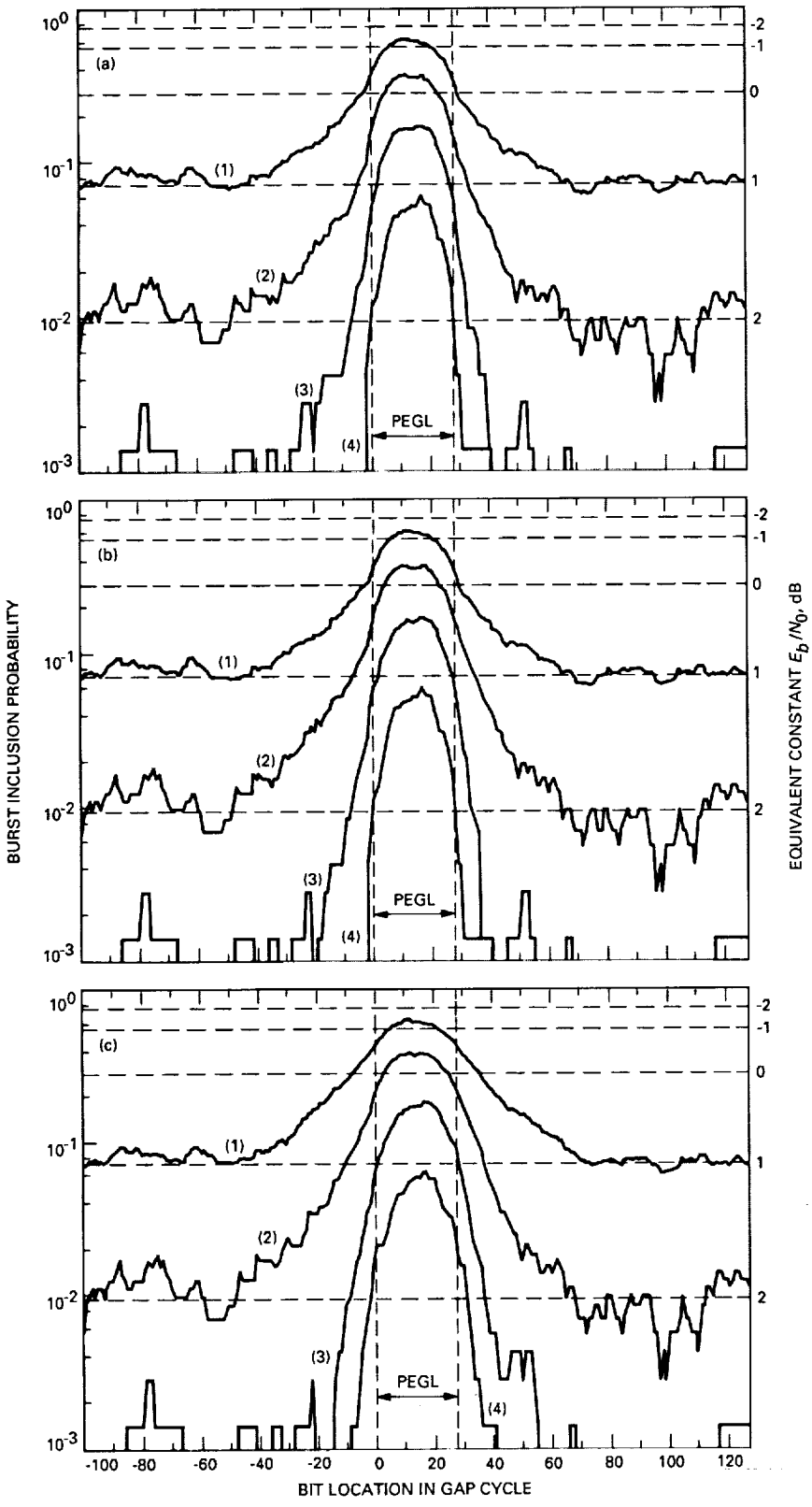


CURVE NO.	E_b/N_0, dB OUTSIDE GAP	E_b/N_0, dB INSIDE GAP
(1)	1	$-\infty$
(2)	2	$-\infty$
(3)	3	$-\infty$
(4)	4	$-\infty$
(5)	∞	$-\infty$

CURVE NO.	E_b/N_0, dB OUTSIDE GAP	E_b/N_0, dB INSIDE GAP
(1)	1	$-\infty$
(2)	2	$-\infty$
(3)	3	$-\infty$
(4)	4	$-\infty$
(5)	∞	$-\infty$

CURVE NO.	E_b/N_0, dB OUTSIDE GAP	E_b/N_0, dB INSIDE GAP
(1)	1	$-\infty$
(2)	2	$-\infty$
(3)	3	$-\infty$
(4)	4	$-\infty$
(5)	∞	$-\infty$

Fig. 1. Decoder error-cycle curves for VLA stand-alone system using (a) optimum branch-metric scaling, (b) constant noise-level scaling, and (c) constant signal-level scaling. (PEGL = predicted effective gap length.)

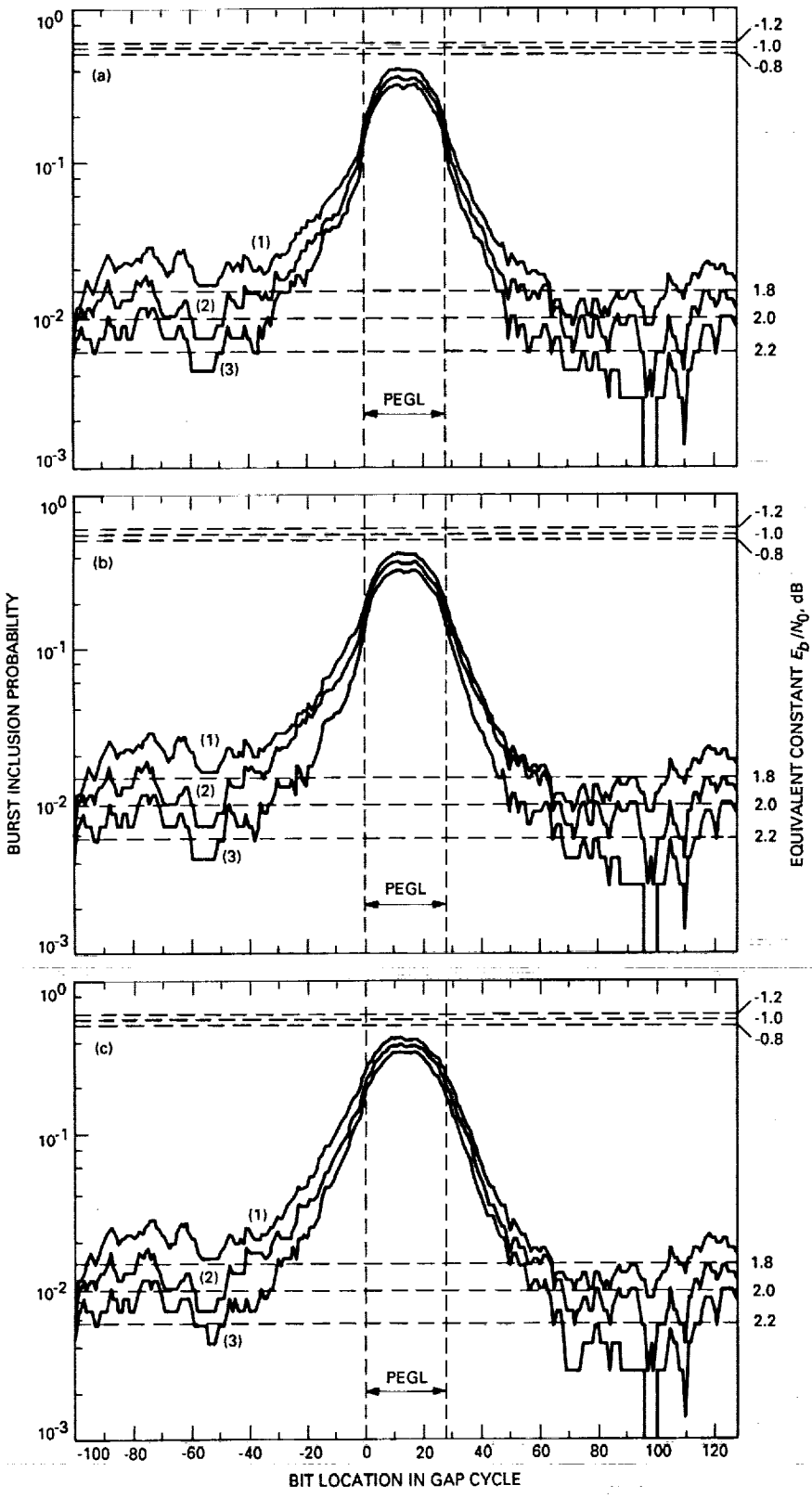


CURVE NO.	E_b/N_0, dB OUTSIDE GAP	INSIDE GAP
(1)	1	−2
(2)	2	−1
(3)	3	0
(4)	4	1

CURVE NO.	E_b/N_0, dB OUTSIDE GAP	INSIDE GAP
(1)	1	−2
(2)	2	−1
(3)	3	0
(4)	4	1

CURVE NO.	E_b/N_0, dB OUTSIDE GAP	INSIDE GAP
(1)	1	−2
(2)	2	−1
(3)	3	0
(4)	4	1

Fig. 2. Decoder error-cycle curves for VLA arrayed equally with Goldstone (3-dB gaps) using (a) optimum branch-metric scaling, (b) constant noise-level scaling, and (c) constant signal-level scaling. (PEGL = predicted effective gap length.)

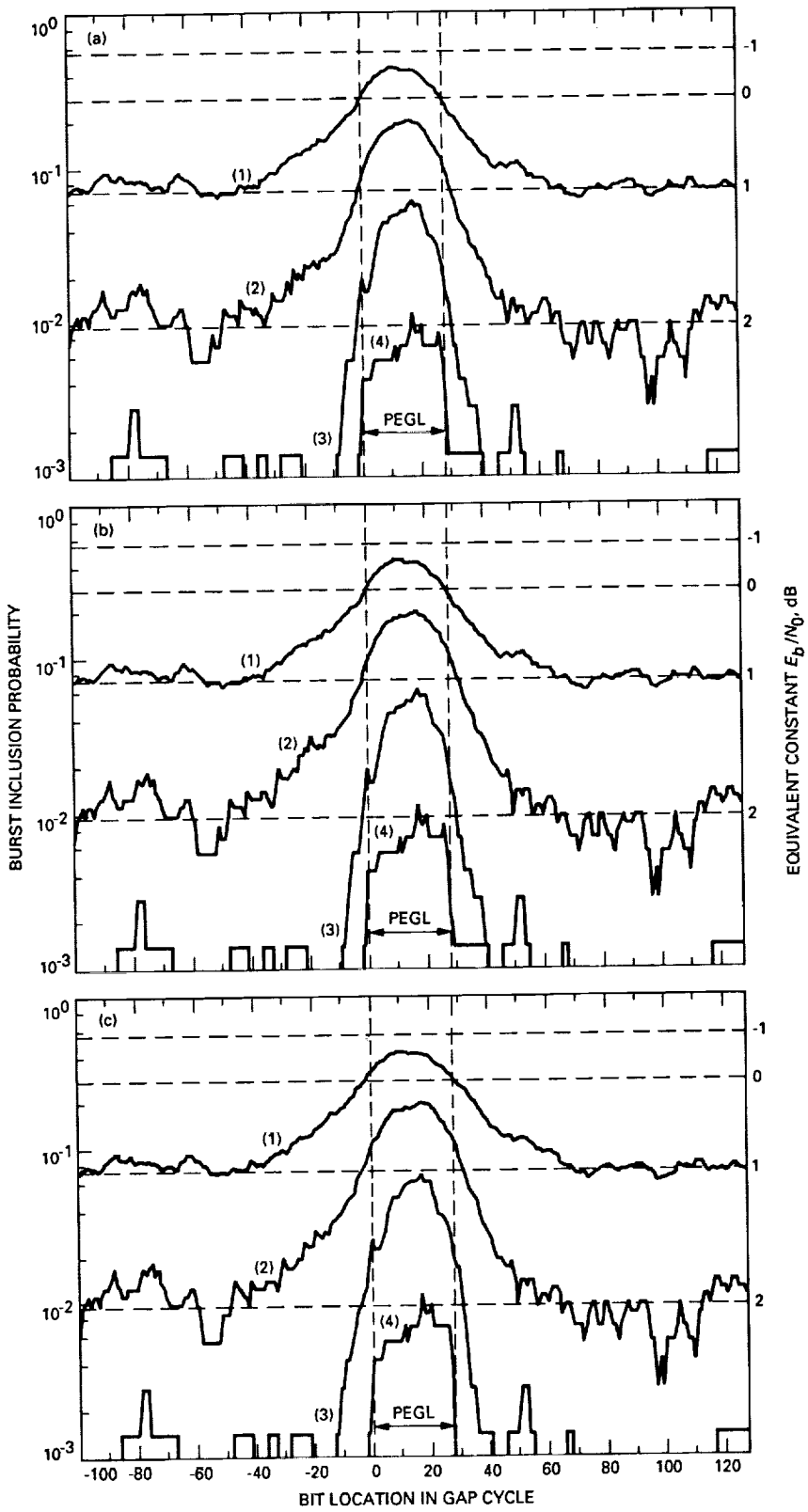


CURVE NO.	E_b/N_0, dB OUTSIDE GAP	E_b/N_0, dB INSIDE GAP
(1)	1.8	−1.2
(2)	2.0	−1.0
(3)	2.2	−0.8

CURVE NO.	E_b/N_0, dB OUTSIDE GAP	E_b/N_0, dB INSIDE GAP
(1)	1.8	−1.2
(2)	2.0	−1.0
(3)	2.2	−0.8

CURVE NO.	E_b/N_0, dB OUTSIDE GAP	E_b/N_0, dB INSIDE GAP
(1)	1.8	−1.2
(2)	2.0	−1.0
(3)	2.2	−0.8

Fig. 3. Decoder error-cycle curves (outside gaps 1.8–2.2 dB) for VLA arrayed equally with Goldstone (3-dB gaps) using (a) optimum branch-metric scaling, (b) constant noise-level scaling, and (c) constant signal-level scaling. (PEGL = predicted effective gap length.)

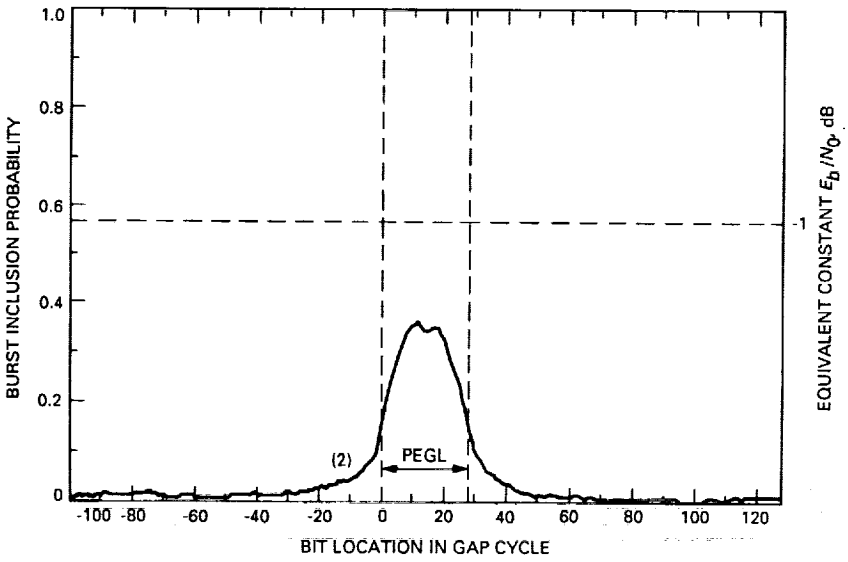


CURVE NO.	E_b/N_0, dB OUTSIDE GAP	INSIDE GAP
(1)	1	−1
(2)	2	0
(3)	3	1
(4)	4	2

CURVE NO.	E_b/N_0, dB OUTSIDE GAP	INSIDE GAP
(1)	1	−1
(2)	2	0
(3)	3	1
(4)	4	2

CURVE NO.	E_b/N_0, dB OUTSIDE GAP	INSIDE GAP
(1)	1	−1
(2)	2	0
(3)	3	1
(4)	4	2

Fig. 4. Decoder error-cycle curves for VLA arrayed unequally with Goldstone (2-dB gaps) using (a) optimum branch-metric scaling, (b) constant noise-level scaling, and (c) constant signal-level scaling. (PEGL = predicted effective gap length.)



CURVE NO.	E_b/N_0, dB OUTSIDE GAP	E_b/N_0, dB INSIDE GAP
(2)	2	−1

Fig. 5. Decoder error-cycle curves for VLA arrayed equally with Goldstone (3-dB gaps) using optimum branch-metric scaling. (PEGL = predicted effective gap length.)