A Node and Frame Synchronization Scheme in the Big Viterbi Decoder Based on Channel Symbol Measurements

K.-M. Cheung and J. I. Statman
Communications Systems Research Section

The Big Viterbi Decoder (BVD), currently under development for the Deep Space Network (DSN), uses three separate algorithms to acquire and maintain node and frame synchronization. The first measures the number of decoded bits between two consecutive renormalization operations (renorm rate), the second detects the presence of the frame marker in the decoded bit stream (bit correlation), while the third searches for an encoded version of the frame marker in the encoded input stream (symbol correlation). This article gives a detailed account of the operation, as well as an evaluation of performance, of the third scheme.

I. Introduction

The present NASA standard concatenated code uses a (7,1/2) convolutional code as its inner code and an 8-bit (255,223) Reed-Solomon (RS) code as its outer code. This system achieves a bit error rate (BER) of 10^{-6} at a bit signal-to-noise ratio (SNR) of 2.53 dB. Recent code search efforts [1,2] show that an improvement of up to 2 dB is possible using a constraint length 15 convolutional code as the inner code. To demonstrate this performance improvement, an experimental (15,1/4) convolutional encoder was implemented on the Galileo spacecraft and a programmable convolutional decoder, the Big Viterbi Decoder (BVD), is currently under development for codes with constraint lengths up to 15 and code rates of $1/N$, $N = 2, 3, \ldots, 6$ [3]. Good node and frame synchronization schemes are essential to realizing the aforementioned performance of the long constraint length codes.

The Maximum-Likelihood Decoder (MCD), manufactured by Linkabit and currently used by the Deep Space Network (DSN), utilizes the state metrics growth rate to acquire and maintain node synchronization. The frame sync procedure, on the other hand, is performed using a 32-bit sync-marker sequence, convolutionally encoded and transmitted from the spacecraft at the beginning of every data frame, consisting of around 10^4 bits. On the ground, the decoded bit stream is monitored for a 32-bit window of agreements with the marker, and a likely sync location is identified by comparing the number of disagreements with a preselected threshold. In addition, a frame-to-frame verification strategy is employed to definitively declare frame sync acquisition or sync loss.

The BVD uses three separate algorithms to acquire and maintain node and frame synchronization. An overview of these algorithms is reported in [4]. Scheme 1 measures the number of decoded bits between two consecutive renormalizations (renorm rate). It is a continuous operation and assumes no knowledge of the data format of the bit streams. This method is used for node sync only. Schemes 2 and 3 require that an a priori known bit pattern, the frame marker, be inserted in regular intervals, known as frames, into the bit stream. Scheme 2 detects the presence of the frame marker in the decoded bit stream (bit correlation), while Scheme 3 searches for an encoded version of the frame marker in the encoded input stream (symbol correlation). These two methods are used for both node sync and frame sync.

The three schemes have their respective advantages and disadvantages. The approach taken in the BVD is to use a combination of all three methods so that overall system performance is optimized. Previous research [5] indicates that each algorithm generates sync statistics that might be more reliable than others in a certain operational environment (e.g., SNR, frame size, operational mode). The overall node and frame sync performances depend on the selection of the right combination of algorithms under the given operational environment.

The synchronization algorithms operate in two modes: the acquisition mode when the decoder is first activated or when a loss of sync is detected, and the tracking mode when the decoder acquires sync and tries to maintain it. In the acquisition mode, a statistic x is measured at all possible locations of the frame marker, and the observed values of x are used to identify likely locations of the marker. Frame-to-frame verification (as in Scheme 2) or integration over several frames (as in Scheme 3) can be used to reduce the probability of miss and the probability of false alarm at the expense of a longer acquisition time. In the tracking mode, the statistic x is monitored from frame to frame at the selected marker location to verify continued sync (flywheeling).

This article gives a detailed account of the operation and an evaluation of the performance of Scheme 3. The concept of node and frame synchronization based on channel symbol measurements is explained in Section II. A performance analysis is given in Section III, and concluding remarks are given in Section IV.

The performance analysis of Scheme 3 indicates that the present 32-bit frame sync marker might not be good enough to meet present and future DSN requirements due to the use of long constraint length codes, as well as a low operating SNR. A 64-bit sync marker should also be considered.

II. Symbol Correlation Scheme

To describe the node and frame sync scheme based on channel symbol measurements, the notations used in [5] are employed. The incoming data bit stream b_i includes both true data bits and sync marker bits λ_i. To simplify subsequent discussion, the coded sync pattern is assumed to have the ideal autocorrelation property. The data bit stream is packaged into data frames $b_i, i = 1, \ldots, B$ of B bits each, and L sync marker bits $\lambda_i, i = 1, \ldots, L$ are included in every data frame. The data bit stream is convolutionally encoded by a rate $1/N$, constraint length K convolutional encoder. The encoded channel symbol stream $s_i, i = 1, \ldots, S$ is likewise partitioned into frames of $S = NB$ symbols each, and each frame includes a set of $M = N(L - K + 1)$ sync marker symbols $m_i, i = 1, \ldots, M$ that are totally determined by the sync marker bits λ_i, $i = 1, \ldots, L$. The remaining $N(B - L + K - 1)$ symbols in each frame are dependent solely on the true data bits or else on a combination of true data bits and sync marker bits. The system diagram introducing the notations for describing Scheme 3 is given in Fig. 1.

The channel symbols are assumed to have constant magnitude s (i.e., $s_i = \pm s$) and are received in additive white Gaussian noise (AWGN) $n_i, i = 1, \ldots, S$ with zero mean and variance σ^2. The ratio $\rho = s^2/\sigma^2$ is a signal-to-noise ratio parameter. In terms of ρ, the channel symbol SNR is $E_s/N_0 = \rho/2$, and the bit SNR is $E_b/N_0 = N\rho/2$. The received symbols $r_i, i = 1, \ldots, S$ are passed through a maximum-likelihood convolutional decoder (Viterbi decoder) to obtain the decoded bits $d_i, i = 1, \ldots, B$.

Two factors contribute to the difficulties of synchronization using channel symbols. First, long constraint length, low-rate codes are designed to operate in a low-SNR environment in which the channel symbols are severely corrupted by noise. Second, an inherent drawback of this scheme is that only $L - K + 1$ bits of frame marker are usable in the correlation. The encoded symbols corresponding to the other $K - 1$ bits depend on the previous contents of the encoder shift register unrelated to the sync marker. To recover enough SNR for the correlation, integration over j ($j \geq 1$) frames is needed at a low SNR and/or when L is small. This is equivalent to increasing the SNR by a factor of j in a one-frame symbol correlation. The sync time, however, is also increased by a factor of j.

Let x be the symbol correlation statistic defined by[1]

$$x = \sum_{i=1}^{M} m_i r_i$$

[1] This "positive correlation" statistic differs slightly from the "negative correlation" statistic used in [5], but it is equivalent in performance effects and corresponds to the actual statistic measured by the BVD.

When the decoder is first activated, or when a loss of sync is detected, the sync system initiates the acquisition mode. The statistic x is measured at all S possible locations of the frame marker, each integrated over j_1 frames, and the observed values of x are compared to a programmable threshold θ_1 to identify likely locations of the marker. The statistic x is said to pass the threshold test if $x \geq \theta_1$. Sync is declared tentatively if only one value of x passes the threshold test and the rest fail. If all S values fail the threshold test, or when two or more values of x pass the threshold test, the decoder aborts the sync process and starts a new sync search again.

After a successful tentative sync declaration, the chosen sync location is subjected to confirmation before the acquisition selection is finalized. The confirmation procedure has not yet been determined and is not addressed here.

After the decoder has acquired sync, the sync system initiates the tracking mode. The statistic x is tested against a preselected threshold θ_2 only at the presumed marker position integrated over j_2 frames. Again, x is said to pass the threshold test if $x \geq \theta_2$. If x fails the threshold test, the out-of-sync hypothesis is declared and the sync system switches to acquisition mode.

The BVD actually has the capability of using two thresholds, $\theta_{1,H}$ and $\theta_{1,L}$ (acquisition mode) or $\theta_{2,H}$ and $\theta_{2,L}$ (tracking mode), where $\theta_{1,L} = -\theta_{1,H}$ and $\theta_{2,L} = -\theta_{2,H}$. The lower threshold $\theta_{1,L}$ or $\theta_{2,L}$ is used to identify the sync marker in the case of bit inversion, which results from the operation of the telemetry receiver and cannot be easily overcome. In this article, only single thresholds θ_1 (acquisition mode) and θ_2 (tracking mode) in the normal non-bit-inversal case are considered. The results are easily extendable to the case of double thresholds.

The general performance expressions in this article are derived for arbitrary combinations of the parameters K, N, L, M, B, and S. In this article, the aforementioned constraint length $K = 15$ convolutional code is assumed, with a code rate $1/N$ of either 1/4 or 1/6. The marker sequence length L is either 32 bits or 64 bits, and the data frame length B is either 5120 or 10,000 bits.

III. Performance Analysis

Let x be integrated over j frames. The observed value of this statistic should be near jMs^2 if $r_i, i = 1, \ldots, M$ contains the marker, and otherwise should be near 0. It is therefore natural to compare the observed values of x against a preselected threshold θ to make tentative yes–no decisions about the location of the marker, according to whether x falls below or exceeds θ. That is

$$x \quad \underset{out-of-sync}{\overset{in-sync}{\gtrless}} \quad \theta$$

If x falls below θ when measured at the true position of the marker, the tentative decision rule results in the true marker location being missed. Conversely, if x does not fall below θ when not measured at the true marker position, then the decision causes a false detection of sync or false alarm. The effectiveness of the sync system can be determined by two competing measurements: the probability of miss (P_M) and the probability of false alarm (P_F), which are both functions of θ and are defined as

$$\begin{aligned} P_M = \text{Prob}[x \geq \theta | &\text{sync marker in the current} \\ &M - \text{symbol window}] \end{aligned}$$

and

$$\begin{aligned} P_F = \text{Prob}[x < \theta | &\text{sync marker not in the current} \\ &M - \text{symbol window}] \end{aligned}$$

Measuring x at the marker after integrating over j frames, the channel symbol correlation statistic x can be expressed as

$$x = \sum_{i=1}^{M} m_i(jm_i + n_i^{(j)}) = \sum_{i=1}^{M} u_i$$

where $jm_i + n_i^{(j)}$ is the ith received symbol integrated over j frames, $n_i^{(j)}$ is $N(0, j\sigma^2)$, u_i is $N(js^2, js^2\sigma^2)$, and x/σ^2 is $N(jMs^2/\sigma^2, (jMs^2)/\sigma^2) = N(jM\rho, Mj\rho)$. Thus, the miss probability P_M is calculated simply as

$$P_M(\theta, j\rho) = Q\left(\frac{jM\rho - \theta/\sigma^2}{\sqrt{Mj\rho}}\right)$$

Away from the marker, the correlation statistic x is a sum of conditionally Gaussian random variables, some with zero mean and some with nonzero mean

$$\begin{aligned} x &= \sum_{i=1}^{M-w} (js^2 + m_i n_i^{(j)}) + \sum_{i=1}^{w} (-js^2 + m_i n_i^{(j)}) \\ &= \sum_{i=1}^{M-w} u_i + \sum_{i=1}^{w} v_i \end{aligned}$$

where $i = 1, \ldots, w$ is the index of the encoded symbols that differ from the marker symbols, v_i is $N(-js^2, js^2\sigma^2)$, and x/σ^2 is $N\{[j(M-2w)s^2]/\sigma^2, (jMs^2)/\sigma^2\} = N[j(M-2w)\rho, jM\rho]$. The false alarm probability is obtained by averaging the conditional Gaussian probability distribution for x over the discrepancy weight distribution Prob$[w]$

$$P_F(\theta, j\rho) = \sum_{w=0}^{M} \text{Prob}[w] Q\left(\frac{2wj\rho - jM\rho + \theta/\sigma^2}{\sqrt{jM\rho}}\right)$$

where Prob$[w] \approx 2^{-M}\binom{M}{w}$ [6]. Note that both $P_M(\theta, j\rho)$ and $P_F(\theta, j\rho)$ depend on the symbol SNR ρ and the integration interval j only in terms of the product $j\rho$, which is the effective symbol SNR after integration over j frames.

Figures 2 and 3 show $P_M(\theta, j\rho)$ versus $P_F(\theta, j\rho)$ for the symbol correlation statistic (frame marker length $L =$ 32 bits) for the (15,1/4) and (15,1/6) codes, respectively. Note that doubling the integration time is equivalent to increasing the SNR by a factor of 2. Thus, for example, integrating over 4 frames at 0.0 dB has the same performance as integrating over 2 frames at 3.0 dB.

A. Acquisition Mode

In this mode, the statistic x is checked at all possible locations of the frame sync marker after integration over j_1 frames. The observed values of x are compared against a pre-set threshold θ_1. Sync is declared if only one of the S values of x passes the threshold test ($x \geq \theta$) and the rest fail ($x < \theta$). With the assumption that the statistics of x are independent from location to location, P_M and P_F (as functions of threshold θ_1 and integration time j_1 frames) determine the probability of acquisition P_{acq} integrated over j_1 frames as follows:

$$P_{acq}(\theta_1, j_1\rho) = [1 - P_F(\theta_1, j_1\rho)]^{S-1}[1 - P_M(\theta_1, j_1\rho)]$$

An optimal normalized threshold $\theta_1/j_1\sigma^2$ that maximizes $P_{acq}(\theta_1, j_1\rho)$ can be evaluated for any given combination of SNR, code rate, frame length, frame marker size, and integration time. Tables 1–4 give optimal normalized thresholds $\theta/j_1\sigma^2$ for the (15,1/4) code. Figures 4(a)–(d) show the respective optimal P_{acq} versus SNR. Tables 5–8 show the optimal normalized thresholds for the (15,1/6) code, and Figs. 5(a)–(d) give the respective optimal P_{acq} versus SNR.

B. Tracking Mode

In this mode, the statistic x is integrated over j_2 frames and then compared against a pre-set threshold θ_2 only at the presumed frame marker position (flywheeling). If $x > \theta_2$, the in-sync hypothesis is assumed, otherwise the decoder declares out of sync. A false declaration of loss of sync during a track when the decoder is actually in sync is a severe offense that causes loss of valuable data and should be avoided. However, when the decoder truly loses sync, it should detect it and correct for it as soon as possible. A general DSN requirement during tracking is no false loss of node sync for 24 hours. For Galileo data rates, 134.4 kbits/sec and 115 kbits/sec, and a frame size of 5120 bits, the DSN requirement can be guaranteed by keeping the probability of false declaration of loss of sync, $P_M(\theta_2, j_2\rho)$, below 4.4×10^{-7}. Tables 9 and 10 give the normalized thresholds $\theta_2/j_2\sigma^2$ for the (15,1/4) code that produce a false loss-of-sync probability of 10^{-7}, with respective frame marker lengths of 32 bits and 64 bits. Tables 11–14 give the corresponding probability of detecting true loss of sync during a track, $1 - P_F(\theta_2, j_2\rho)^{n/j_2}$, versus the number of frames n to detect it. Tables 15 and 16 give the normalized thresholds for the (15,1/6) code, and Tables 17–20 give the corresponding probability of detecting true loss of frame versus the number of frames n to detect it.

IV. Conclusion

Some important conclusions can be derived from the performance analysis in the previous section. In the acquisition mode, in order to achieve a 99 percent probability of acquisition with a 32-bit frame marker, integration over 12 to 16 frames is required. This corresponds to between 0.5 and 1 sec of acquisition time at the Galileo code rates, if only the channel symbol correlation method is used. With a 64-bit marker, the same probability of acquisition (99 percent) can be achieved by integrating over four frames or less. In the tracking mode, a 10^{-7} probability of false declaration of out of sync is imposed to satisfy the DSN requirement of no loss of node sync for 24 hours. A 95 percent probability of detecting true out of sync with a 32-bit marker can be achieved by integrating over eight or more frames. With a 64-bit marker, less than two frames are required.

References

[1] J. H. Yuen and Q. D. Vo, "In Search of a 2-dB Coding Gain," *TDA Progress Report 42-83*, vol. July–September 1985, Jet Propulsion Laboratory, Pasadena, California, pp. 26–33, November 15, 1985.

[2] S. Dolinar, "A New Code for Galileo," *TDA Progress Report 42-93*, vol. January–March 1988, Jet Propulsion Laboratory, Pasadena, California, pp. 83–96, May 15, 1988.

[3] J. Statman, G. Zimmerman, F. Pollara, and O. Collins, "A Long Constraint Length VLSI Viterbi Decoder for the DSN," *TDA Progress Report 42-95*, vol. July–September 1988, Jet Propulsion Laboratory, Pasadena, California, pp. 134–142, November 15, 1988.

[4] J. Statman, B. Siev, J. Rabkin, and K.-M. Cheung, "Node and Frame Synchronization in the Big Viterbi Decoder," *TDA Progress Report 42-103 (this issue)*, vol. July–September 1990, Jet Propulsion Laboratory, Pasadena, California, pp. 154-160, November 15, 1990.

[5] S. Dolinar and K.-M. Cheung, "Frame Synchronization Methods Based on Channel Symbol Measurements," *TDA Progress Report 42-98*, vol. April–June 1989, Jet Propulsion Laboratory, Pasadena, California, pp. 121–137, August 15, 1989.

[6] K.-M. Cheung, "The Weight Distribution and Randomness of Linear Codes," *TDA Progress Report 42-97*, vol. January–March 1989, Jet Propulsion Laboratory, Pasadena, California, pp. 208–215, May 15, 1989.

Table 1. Optimal normalized acquisition mode thresholds for the (15,1/4) code for $L = 32$ and $B = 5120$

SNR	$j = 4$	$j = 8$	$j = 12$	$j = 16$
0.0	35.00	34.10	33.80	33.80
0.1	35.84	34.84	34.64	34.54
0.2	36.60	35.60	35.40	35.30
0.3	37.47	36.47	36.17	36.07
0.4	38.27	37.27	37.07	36.97
0.5	39.09	38.09	37.89	37.89
0.6	39.93	39.03	38.83	38.73
0.7	40.80	39.90	39.70	39.70
0.8	40.68	40.78	40.58	40.58

Table 2. Optimal normalized acquisition mode thresholds for the (15,1/4) code for $L = 32$ and $B = 10{,}000$

SNR	$j = 4$	$j = 8$	$j = 12$	$j = 16$
0.0	35.30	34.20	33.90	33.80
0.1	36.04	34.94	34.74	34.64
0.2	36.80	35.80	35.50	35.40
0.3	37.67	36.57	36.27	36.27
0.4	38.47	37.37	37.17	37.07
0.5	39.29	38.29	37.99	37.99
0.6	40.13	39.13	38.93	38.83
0.7	41.10	40.00	39.80	39.70
0.8	41.98	40.98	40.68	40.68

Table 3. Optimal normalized acquisition mode thresholds for the (15,1/4) code for $L = 64$ and $B = 5120$

SNR	$j = 2$	$j = 4$
0.0	93.80	91.80
0.1	95.93	93.93
0.2	98.01	96.01
0.3	100.15	98.25
0.4	102.45	100.55
0.5	104.70	102.80
0.6	107.02	105.22
0.7	109.49	107.57
0.8	111.72	109.92

Table 4. Optimal normalized acquisition mode thresholds for the (15,1/4) code for $L = 64$ and $B = 10{,}000$

SNR	$j = 2$	$j = 4$
0.0	94.20	92.00
0.1	96.23	94.13
0.2	98.41	96.31
0.3	100.55	98.45
0.4	102.55	100.75
0.5	105.10	103.00
0.6	107.42	105.42
0.7	109.89	107.42
0.8	112.12	110.32

Table 5. Optimal normalized acquisition mode thresholds for the (15,1/6) code for $L = 32$ and $B = 5120$

SNR	$j = 8$	$j = 12$	$j = 16$
0.0	34.90	34.50	34.40
0.1	35.64	35.34	35.14
0.2	36.50	36.10	36.00
0.3	37.27	36.57	36.77
0.4	38.07	37.77	37.67
0.5	38.99	38.69	38.49
0.6	39.83	39.53	39.43
0.7	40.80	40.40	40.30
0.8	41.68	41.38	41.28

Table 6. Optimal normalized acquisition mode thresholds for the (15,1/6) code for $L = 32$ and $B = 10{,}000$

SNR	$j = 8$	$j = 12$	$j = 16$
0.0	35.00	34.60	34.40
0.1	35.74	35.34	35.24
0.2	36.50	36.10	36.00
0.3	37.27	36.87	36.87
0.4	38.27	37.87	37.67
0.5	39.09	38.69	38.59
0.6	39.93	39.63	39.43
0.7	40.80	40.50	40.40
0.8	41.78	41.48	41.28

Table 7. Optimal normalized acquisition mode thresholds for the (15,1/6) code for $L = 64$ and $B = 5120$

SNR	$j = 2$	$j = 4$
0.0	97.30	95.00
0.1	99.43	97.13
0.2	101.71	99.31
0.3	103.95	101.55
0.4	106.25	103.95
0.5	108.60	106.30
0.6	111.02	108.72
0.7	113.49	111.19
0.8	115.82	113.52

Table 8. Optimal normalized acquisition mode thresholds for the (15,1/6) code for $L = 64$ and $B = 10{,}000$

SNR	$j = 2$	$j = 4$
0.0	97.60	95.20
0.1	99.83	97.33
0.2	99.63	97.13
0.3	104.25	101.75
0.4	106.55	104.15
0.5	109.00	106.50
0.6	111.42	108.92
0.7	113.89	111.49
0.8	116.20	113.82

Table 9. Normalized tracking mode thresholds for the (15,1/4) code to achieve P (false declaration of loss of sync) $= 10^{-7}$ for $L = 32$

SNR	$j = 2$	$j = 4$	$j = 8$
0.0	25.51	28.58	30.76
0.1	26.23	29.34	31.53
0.2	26.97	20.11	32.33
0.3	27.71	30.89	33.14
0.4	28.47	31.70	33.98
0.5	29.28	32.53	34.83
0.6	30.09	33.38	35.71
0.7	30.93	34.26	36.62
0.8	31.78	35.15	37.53

Table 10. Normalized tracking mode thresholds for the (15,1/4) code to achieve P (false declaration of loss of sync) $= 10^{-7}$ for $L = 64$

SNR	$j = 2$	$j = 4$	$j = 8$
0.0	85.52	87.64	91.26
0.1	84.64	89.82	93.49
0.2	86.82	92.06	95.77
0.3	89.05	94.35	98.10
0.4	91.34	96.71	100.50
0.5	93.68	99.11	102.94
0.6	96.09	101.57	105.45
0.7	98.54	104.09	108.02
0.8	100.85	106.47	110.44

Table 11. Probability of detecting true loss of sync in the (15,1/4) code for $L = 32$, integration over two frames

SNR	$n = 2$	$n = 4$	$n = 6$	$n = 8$
0.0	0.2650	0.4598	0.6029	0.7082
0.1	0.2800	0.4817	0.6268	0.7313
0.2	0.2956	0.5038	0.6505	0.7538
0.3	0.3117	0.5262	0.6738	0.7755
0.4	0.3282	0.5487	0.6968	0.7963
0.5	0.3452	0.5712	0.7192	0.8162
0.6	0.3626	0.5937	0.7411	0.8350
0.7	0.3805	0.6162	0.7622	0.8527
0.8	0.3987	0.6384	0.7826	0.8692

Table 12. Probability of detecting true loss of sync in the (15,1/4) code for $L = 32$, integration over four frames

SNR	$n = 4$	$n = 8$
0.0	0.8055	0.9622
0.1	0.8198	0.9675
0.2	0.8334	0.9723
0.3	0.8464	0.9764
0.4	0.8589	0.9801
0.5	0.8705	0.9832
0.6	0.8815	0.9860
0.7	0.8919	0.9883
0.8	0.9017	0.9903

Table 13. Probability of detecting true loss of sync in the (15,1/4) code for $L = 32$, integration over eight frames

SNR	$n = 8$
0.0	0.9939
0.1	0.9948
0.2	0.9955
0.3	0.9962
0.4	0.9962
0.5	0.9973
0.6	0.9977
0.7	0.9981
0.8	0.9984

Table 14. Probability of detecting true loss of sync in the (15,1/4) code for $L = 64$, integration over two frames

SNR	$n = 2$	$n = 4$	$n = 6$	$n = 8$
0.0	0.9714	0.9992	1.0000	1.0000
0.1	0.9756	0.9994	1.0000	1.0000
0.2	0.9793	0.9996	1.0000	1.0000
0.3	0.9825	0.9997	1.0000	1.0000
0.4	0.9853	0.9998	1.0000	1.0000
0.5	0.9878	0.9999	1.0000	1.0000
0.6	0.9898	0.9999	1.0000	1.0000
0.7	0.9916	0.9999	1.0000	1.0000
0.8	0.9931	1.0000	1.0000	1.0000

Table 15. Normalized tracking mode thresholds for the (15,1/6) code to achieve P (false declaration of loss of sync) $= 10^{-7}$ for $L = 32$

SNR	$j = 2$	$j = 4$	$j = 8$
0.0	27.43	29.94	31.72
0.1	28.18	30.71	32.51
0.2	28.94	31.50	33.32
0.3	29.70	32.30	34.14
0.4	30.50	33.13	34.99
0.5	31.32	33.98	35.85
0.6	32.15	34.84	36.74
0.7	33.02	35.74	37.66
0.8	33.89	36.64	38.58

Table 16. Normalized tracking mode thresholds for the (15,1/6) code to achieve *P* (false declaration of loss of sync) = 10^{-7} for *L* = 64

SNR	$j = 2$	$j = 4$	$j = 8$
0.0	85.72	89.91	92.86
0.1	87.89	92.12	95.11
0.2	90.10	94.38	97.41
0.3	92.37	96.70	99.76
0.4	94.70	99.08	102.18
0.5	97.08	101.51	104.64
0.6	99.52	104.00	107.17
0.7	102.02	106.55	109.75
0.8	104.37	108.95	112.19

Table 17. Probability of detecting true loss of sync in the (15,1/6) code for *L* = 32, integration over two frames

SNR	$n = 2$	$n = 4$	$n = 6$	$n = 8$
0.0	0.0800	0.1537	0.2214	0.2837
0.1	0.0861	0.1648	0.2367	0.3025
0.2	0.0926	0.1766	0.2528	0.3220
0.3	0.0995	0.1890	0.2697	0.3423
0.4	0.1068	0.2021	0.2873	0.3634
0.5	0.1145	0.2160	0.3058	0.3853
0.6	0.1228	0.2305	0.3250	0.4079
0.7	0.1315	0.2458	0.3450	0.4311
0.8	0.1408	0.2617	0.3656	0.4549

Table 18. Probability of detecting true loss of sync in the (15,1/6) code for *L* = 32, integration over four frames

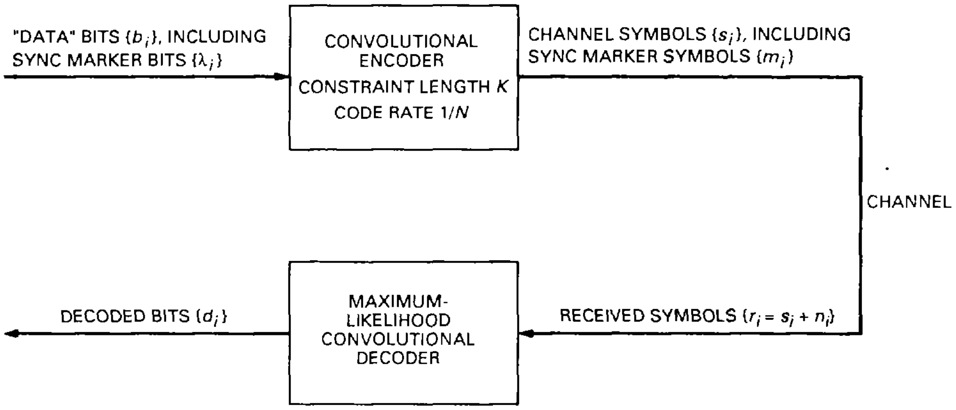
SNR	$n = 4$	$n = 8$	$n = 12$	$n = 16$
0.0	0.4834	0.7332	0.8622	0.9288
0.1	0.5039	0.7539	0.8779	0.9394
0.2	0.5245	0.7739	0.8924	0.9489
0.3	0.5452	0.7931	0.9059	0.9572
0.4	0.5659	0.8115	0.9182	0.9645
0.5	0.5865	0.8290	0.9293	0.9708
0.6	0.6071	0.8456	0.9393	0.9762
0.7	0.6275	0.8612	0.9483	0.9807
0.8	0.6477	0.8759	0.9563	0.9846

Table 19. Probability of detecting true loss of sync in the (15,1/6) code for *L* = 32, integration over eight frames

SNR	$n = 8$
0.0	0.9506
0.1	0.9564
0.2	0.9618
0.3	0.9666
0.4	0.9709
0.5	0.9747
0.6	0.9782
0.7	0.9812
0.8	0.9839

Table 20. Probability of detecting true loss of sync in the (15,1/6) code for *L* = 64, integration over two frames

SNR	$n = 2$	$n = 4$	$n = 6$	$n = 8$
0.0	0.7841	0.9534	0.9899	0.9978
0.1	0.8019	0.9608	0.9922	0.9985
0.2	0.8190	0.9672	0.9941	0.9989
0.3	0.8353	0.9729	0.9955	0.9993
0.4	0.8507	0.9777	0.9967	0.9995
0.5	0.8652	0.9818	0.9976	0.9997
0.6	0.8788	0.9853	0.9982	0.9998
0.7	0.8916	0.9882	0.9987	0.9999
0.8	0.9034	0.9907	0.9991	0.9999


"DATA" BITS $\{b_i\}$, INCLUDING SYNC MARKER BITS $\{\lambda_i\}$
CONVOLUTIONAL ENCODER
CONSTRAINT LENGTH K
CODE RATE $1/N$
CHANNEL SYMBOLS $\{s_i\}$, INCLUDING SYNC MARKER SYMBOLS $\{m_i\}$
CHANNEL
RECEIVED SYMBOLS $\{r_i = s_i + n_i\}$
MAXIMUM-LIKELIHOOD CONVOLUTIONAL DECODER
DECODED BITS $\{d_i\}$

SYNC MARKER BITS $\{\lambda_i, i = 1,..., L\}$ ARE TRANSMITTED ONCE EVERY DATA FRAME $\{b_i, i = 1,..., B\}$

WITHIN EVERY SYMBOL FRAME $\{s_i, i = 1,..., S\}$, THE SYNC MARKER SYMBOLS $\{m_i, i = 1,..., M\}$ ARE THOSE SYMBOLS THAT ARE COMPLETELY DETERMINED BY THE SYNC MARKER BITS

$S = NB \qquad M = N(L - K + 1)$

Fig. 1. Scheme 3.

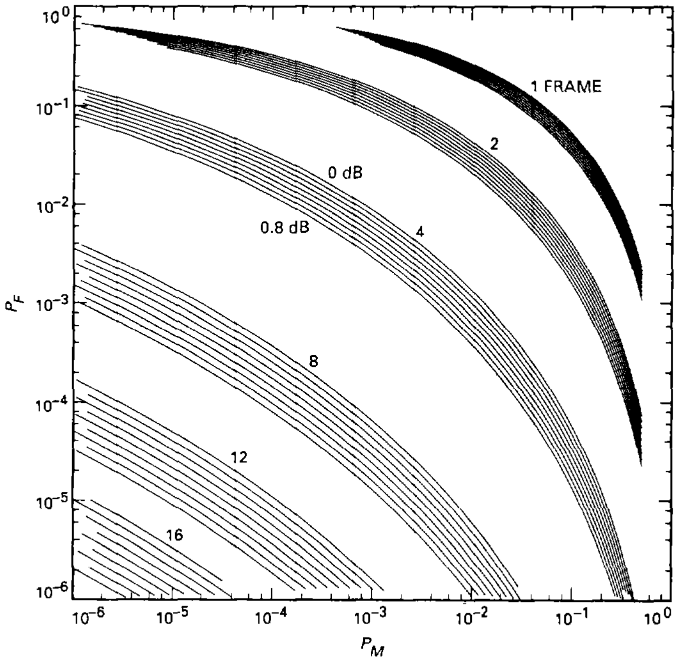


Fig. 2. P_F versus P_M for the (15,1/4) code, $L = 32$.

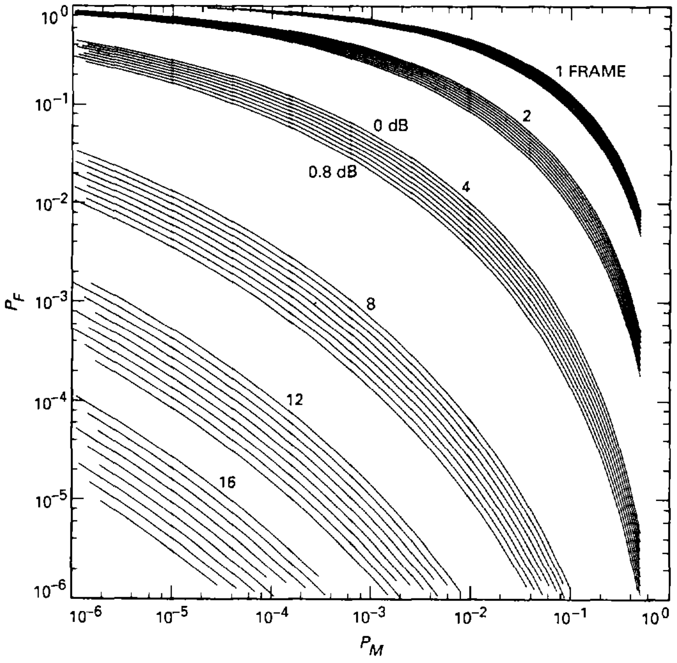


Fig. 3. P_F versus P_M for the (15,1/6) code, $L = 32$.

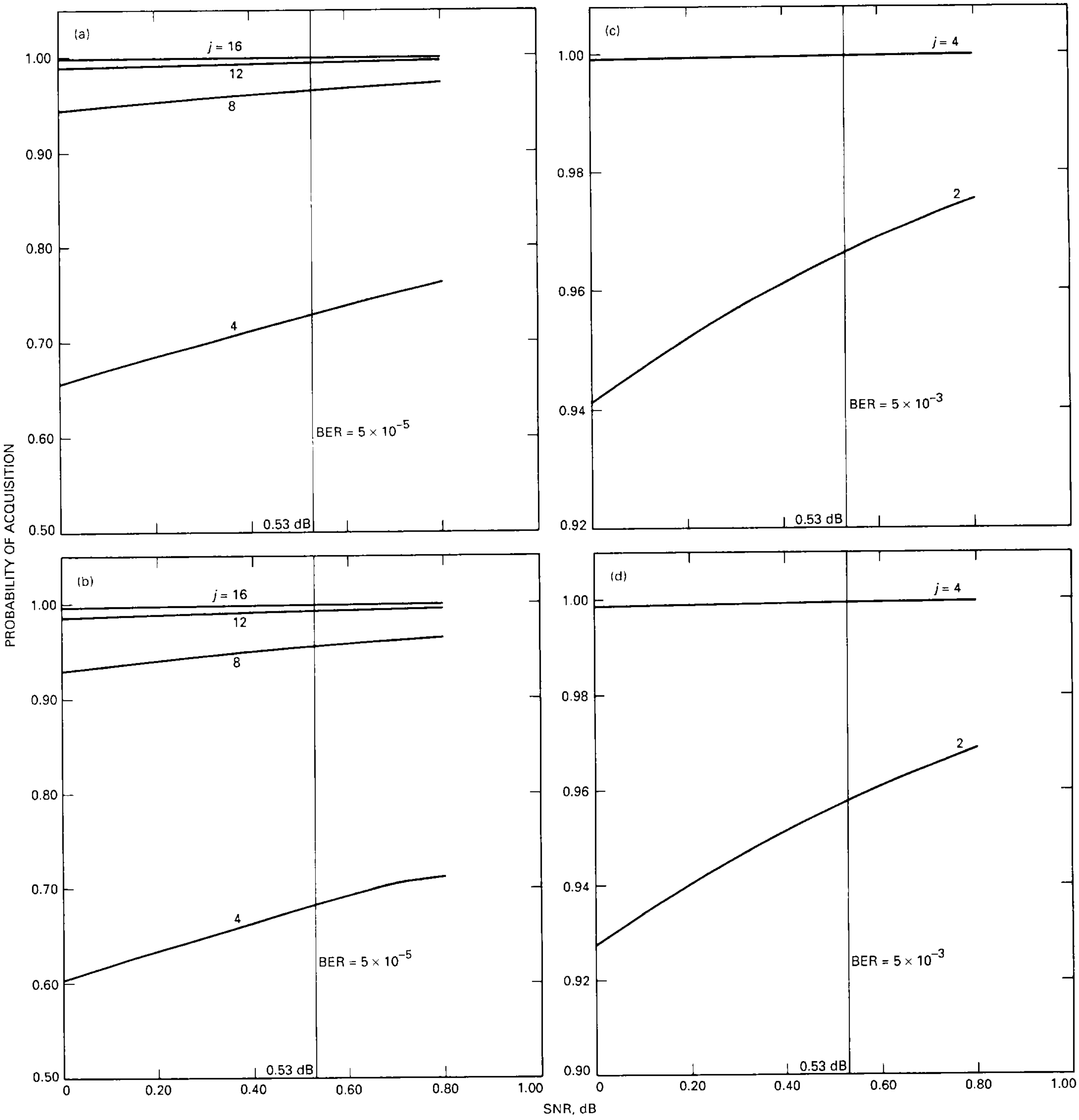


Fig. 4. P_{acq} versus SNR for the (15,1/4) code: (a) $L = 32$ and $B = 5120$; (b) $L = 32$ and $B = 10{,}000$; (c) $L = 64$ and $B = 5120$; and (d) $L = 64$ and $B = 10{,}000$.

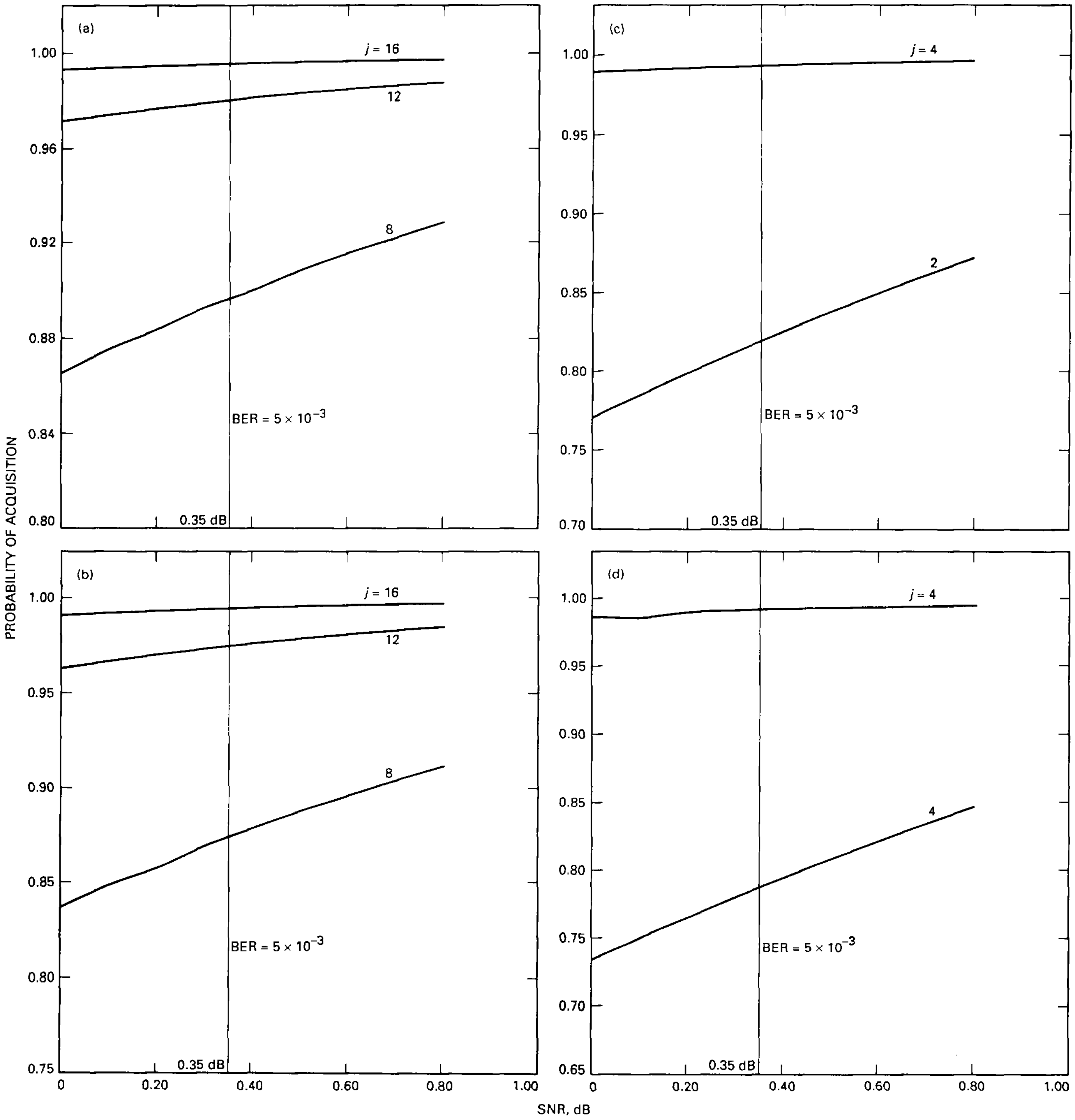


Fig. 5. P_{acq} versus SNR for the (15,1/6) code: (a) $L = 32$ and $B = 5120$; (b) $L = 32$ and $B = 10{,}000$; (c) $L = 64$ and $B = 5120$; and (d) $L = 64$ and $B = 10{,}000$.