

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

Technical Memorandum 33-571

*Telecommunications Systems Design
Techniques Handbook*

*Edited by
R. E. Edelson*

(NASA-CR-129108) TELECOMMUNICATIONS
SYSTEMS DESIGN TECHNIQUES HANDBOOK R.E.
Edelson (Jet Propulsion Lab.) 15 Jul. 1972
448 p
CSCL 17B

N73-11137

G3/07
Unclas
46209

REPRODUCED BY
NATIONAL TECHNICAL
INFORMATION SERVICE
U. S. DEPARTMENT OF COMMERCE
SPRINGFIELD, VA. 22161

JET PROPULSION LABORATORY
CALIFORNIA INSTITUTE OF TECHNOLOGY
PASADENA, CALIFORNIA

July 15, 1972

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

Technical Memorandum 33-571

*Telecommunications Systems Design
Techniques Handbook*

*Edited by
R. E. Edelson*

**JET PROPULSION LABORATORY
CALIFORNIA INSTITUTE OF TECHNOLOGY
PASADENA, CALIFORNIA**

July 15, 1972
1

Technical Memorandum 33-571

Copyright © 1972
Jet Propulsion Laboratory
California Institute of Technology

Prepared Under Contract No. NAS 7-100
National Aeronautics and Space Administration

REPRODUCTION RESTRICTIONS OVERRIDDEN

NASA Scientific and Technical Information Facility

ii

SECTION V

TELEMETRY SYSTEM DESIGN

5.1 INTRODUCTION

The purpose of a telemetry system for interplanetary spacecraft is to transmit to Earth information of scientific and engineering value, and of the physical condition of the spacecraft. It is the goal of the telemetry system designer to produce a spacecraft system design which will provide the required amount of information of a specified quality in the most efficient and cost-effective manner possible. The design must be consistent with the resources of the project and the capabilities and resources of the Deep Space Network for the time period in which the spacecraft will operate. Finally, the design must be based on predictable performance, and the parameters governing the final quality of the data must be carefully analyzed, tested, and controlled.

This section describes the elements of telemetry system design techniques, compatible with the DSN, currently available to the system designer; factors which determine the efficiency of the telemetry system; factors which determine data quality; the type of data required for producing a good telemetry system design; and tradeoff considerations for selection between single and dual subcarrier design.

5.2 DESCRIPTION OF THE TELEMETRY SYSTEM

A telemetry system for interplanetary spacecraft performs the functions described below. It obtains data in the spacecraft from the data handling system in a form ready to be coded and modulated. Data may originally be in analog form from sensors and instruments, or it may be inherently discrete or digital data, such as counter readings, status information, or synchronization codes. This data is usually processed into binary digits whose period and clock rate are accurately known. The digital telemetry system may code any or all of this data into code symbols. Figure 5-1 is a functional block diagram representation of a telemetry system for interplanetary spacecraft.

The telemetry system also performs a modulation function. Under certain circumstances, a project may choose an analog modulation scheme for

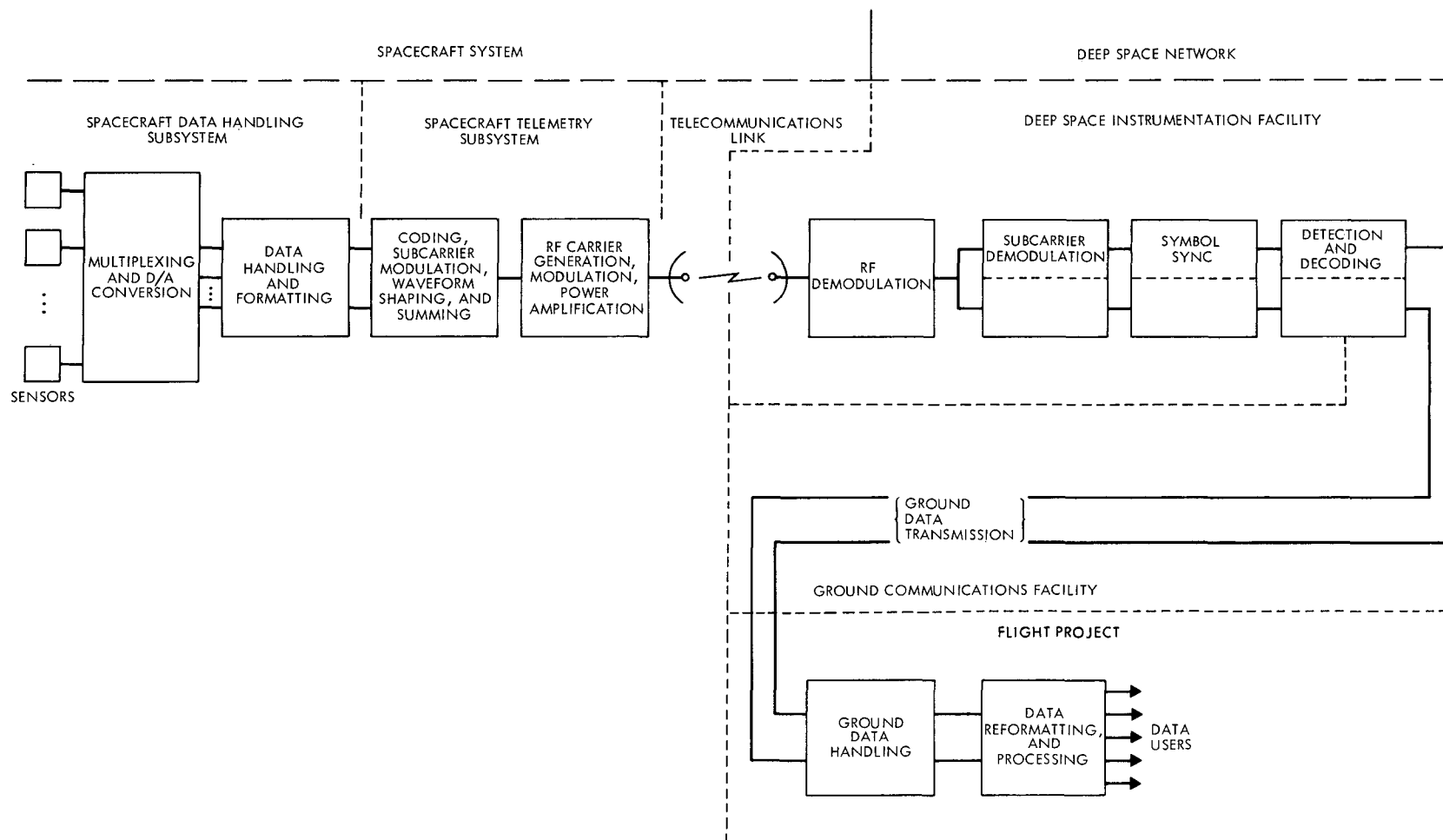


Fig. 5-1. Generalized telemetry system

some of its data. However, the Multi-Mission Telemetry System (MMTS) of the DSN is not equipped to demodulate and process analog data, so the project must supply mission-dependent analog demodulation equipment. The DSN is set up to handle digital telemetry systems, and some choice as to modulation technique is available to the system designer. Digitized data is normally modulated onto one or more subcarriers. Each subcarrier constitutes a data channel. The data-bearing subcarriers are processed to obtain a desired waveform, and are combined into a composite telemetry signal. Since transmission of electromagnetic radiation to Earth is more efficient at microwave frequencies, an rf carrier is generated and the composite telemetry signal modulates the carrier.

The DSN is equipped to receive at S- and X-band (64-meter stations) frequencies, so that the spacecraft may transmit data on two carriers, one in each band. The rf signal is amplified and radiated by an antenna whose gain pattern is designed with the unique requirements of the mission in mind.

On the ground, antennas of the DSN intercept the radiated rf power and amplify the signal, using low-noise masers. Each subcarrier is demodulated by a Subcarrier Demodulator Assembly (SDA). If analog modulation is used by the spacecraft, the project-supplied mission-dependent demodulator obtains its data at this point. If a DSN-compatible modulation technique is used, the SDA will recover data from the signal, and produce a waveform which consists of demodulated data bits, plus noise.

The output from the SDA is fed to the Symbol Synchronizer Assembly (SSA) which integrates the data over a bit time using a matched filter, and detects transitions in the integrated data, to establish symbol sync and to make an estimate of the transmitted symbols.

If no coding was used, the estimated data bits are sent to the Telemetry and Command Data Handling System (TCD) for recording data at the station, and for formatting and subsequent transmission from the tracking station to the Project where the data is decommutated and presented to the data users.

If the data was coded on board the spacecraft, the detected symbols from the SSA are fed to a decoder before processing in the TCD. At present there exists a Block Decoder Assembly (BDA) for decoding (32, 6) and (16, 5)

Biorthogonal Block Codes, and a Data Decoder Assembly (DDA) which decodes convolutionally encoded data. The decoded data is then presented to the TCD for recording and transmitting to the project.

5.2.1 Modulation Techniques for Digital Communications

Demodulation capability is presently available at the DSS for signals whose rf carrier and subcarriers are phase modulated. A phase-lock-loop receiver is employed by the MMTS to obtain a phase reference so that the carrier is coherently demodulated.

The MMTS can accommodate signals utilizing either square- or sine-wave subcarriers. Normally one or two simultaneous subcarriers are used. The number is limited by the number of SDA's and SSA's available in the stations.

5.2.1.1 Phase Shift Keyed Modulation. Figure 5-2 shows a schematic representation of a spacecraft telemetry system using multiple subcarriers which phase modulate an rf carrier. In other words, the signal transmitted by the spacecraft is of the form

$$y(t) = \sqrt{2P_t} \sin \left\{ \omega_c t + \phi_o + \theta_1 (D_1 \oplus SC_1) + \theta_2 (D_2 \oplus SC_2) + \dots \right\} \quad (5.2-1)$$

where

$y(t)$ = Transmitted waveform

P_t = Transmitted power at the transmitting antenna terminals

ω_c = Carrier frequency

ϕ_o = Phase reference, a random variable, uniformly distributed between 0 and 2π

θ_i = Modulation index (degrees) for the i^{th} subcarrier channel

D_i = Binary data symbols in the i^{th} channel

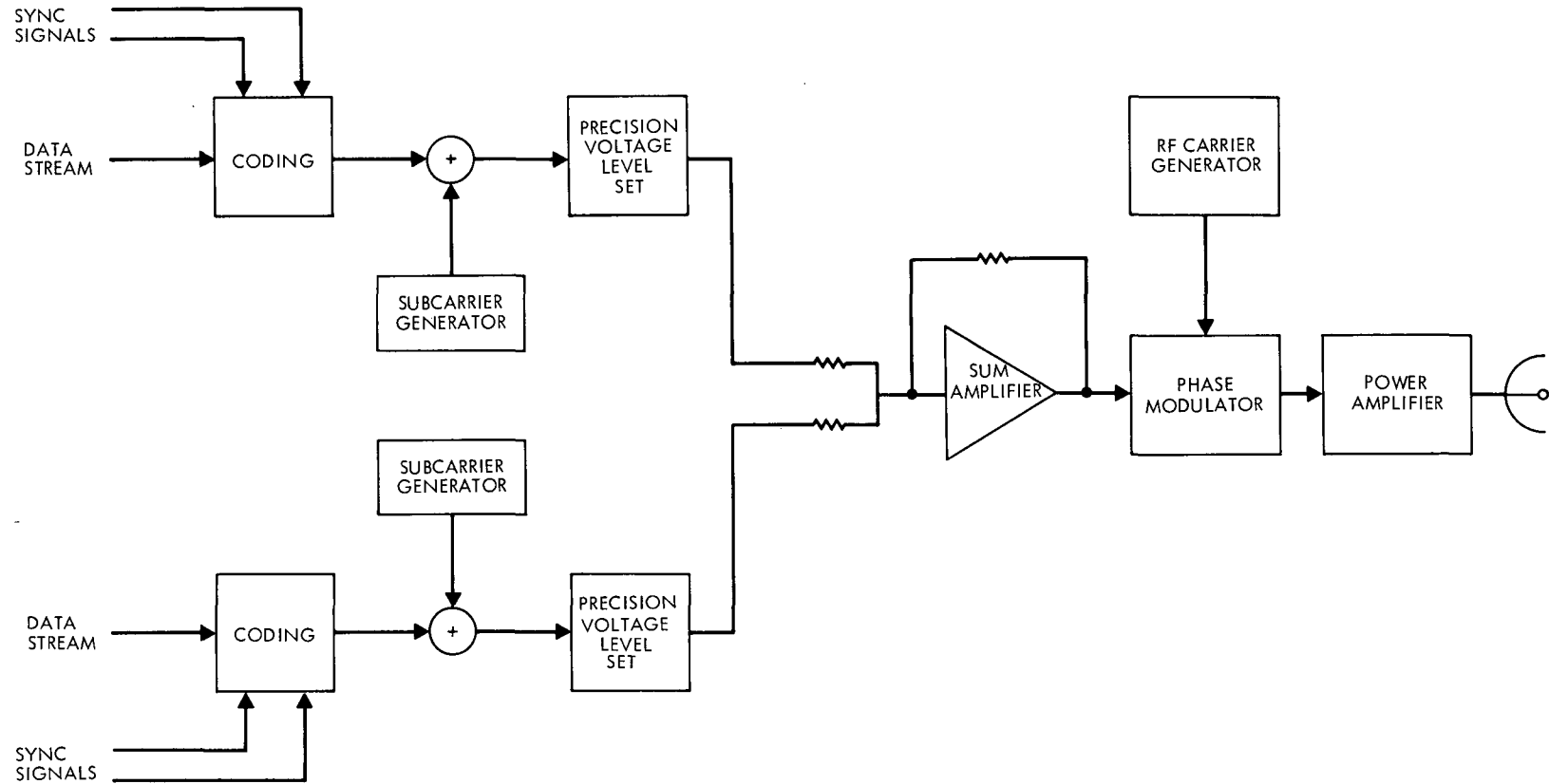


Fig. 5-2. Spacecraft telemetry subsystem

SC_i = Subcarrier waveform in the i^{th} channel

$\oplus$ = Modulo 2 addition for square-wave subcarriers and multiplication for sine waves.

The phase modulator is a device which shifts the phase of the rf carrier in linear proportion to the voltage applied by the composite telemetry waveform. The modulation indices, θ_i , are defined by

$$\theta_i \triangleq T_i K \quad (5.2-2)$$

where

θ_i = Modulation index (degrees)

T_i = Peak voltage level of the i^{th} telemetry channel (volts)

K = Phase modulator sensitivity (degrees per volt)

Figure 5-3 shows how the phase-lock receiver derives a carrier reference by which the subcarrier and data may be demodulated from the received waveform. The received signal, $x(t)$, is mixed with a monochromatic signal of the estimated frequency of $x(t)$. The output of the mixer consists of a waveform having components of the form $\cos(\phi - \hat{\phi})$ and $\cos(2\omega_c t + \phi + \hat{\phi})$. The loop filter is a low-pass filter which removes the double frequency terms to produce a slowly varying error signal which changes the output of a voltage controlled oscillator (VCO) in such a way that the phase of the VCO output $\hat{\phi}$ tries to match the average phase ϕ of $x(t)$. In the absence of noise, if the free-running frequency of the VCO is identical to $x(t)$ and if the changes in $\phi(t)$ due to modulation are of a frequency very much greater than the loop bandwidth of the receiver, $\hat{\phi}$ will equal ϕ , and the carrier reference from the VCO will be coherent with the phase of the received carrier.

Practical receivers will introduce some white gaussian noise in the first amplifier before the receiver, so that the phase of $x(t)$ is corrupted by additive noise of one-sided spectral density N_o (watts/Hz). The phase estimate $\hat{\phi}$ generated by the receiver will be an imperfect replica of the phase of the received rf carrier. The receiver in this case is said to be partially coherent.

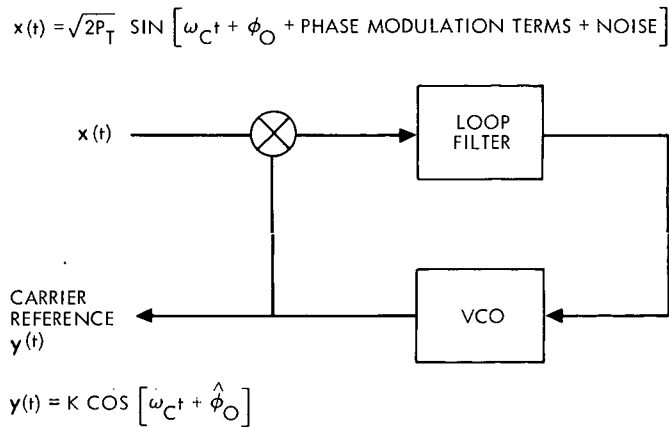


Fig. 5-3. Carrier tracking phase lock loop

More details on the operation of the phase locked loop receiver may be found in references 5-1 and 5-2 and in section IV.

The received signal, $x(t)$, is demodulated as follows: The signal $x(t)$ is multiplied by the carrier reference derived by the phase lock loop (PLL) to obtain the waveform

$$x(t) \cos (\omega_c t + \hat{\phi})$$

where

$$\hat{\phi} = \text{Estimate of the phase reference } \phi_o.$$

The components about $2\omega_c$ are filtered out, leaving

$$\sin \left\{ \phi + \theta_1 (D_1 \oplus SC_1) + \theta_2 (D_2 \oplus SC_2) + \dots \right\} \quad (5.2-3)$$

where

$$\phi = \phi_o - \hat{\phi}$$

The frequencies of the subcarrier SC_1 , SC_2 , etc. must be sufficiently far apart so that the terms $D_i \oplus SC_i$ each occupy different portions of the frequency spectrums. The data-bearing subcarriers can each be isolated by an SDA, and each subcarrier removed from its data. For convenience in circuit

design, the process of removing the carrier reference is performed in the SDA after removal of the subcarriers. Conceptually, the procedure for removal of the subcarriers is the same as that described above. Figure 5-4 gives a schematic representation of the entire demodulation process.

5.3 CHANNEL QUALITY

Of prime concern in the design of any telemetry system is the quality of the data which is returned to the user. It is the responsibility of the telemetry system designer to obtain the requirements for data quality from each user in his project, and thus determine design point values of the appropriate parameters of his system.

The quality of analog signals can usually be expressed in terms of the mean square error of the signal immersed in noise. The quality of digital data is usually expressed in terms of either the data bit error rate or the data word error rate. The choice of bit or word error rate as a criterion will depend primarily on the type and use of the data.

The parameters appropriate for specifying the quality of data depend on the type of communication channel and the modulation techniques applied. For the free-space channel usually encountered by deep space vehicles, the channel is characterized as Gaussian, and the noise (which is almost entirely generated in the front end of the receiver) is white, additive noise characterized by some constant noise spectral density, N_0 watts/Hz.

For other types of channels occasionally encountered by deep space vehicles, such as a multipath channel, the characterization is other than Gaussian. It may be Rician, or in the extreme, a non-parametric characterization may be necessary.

We will primarily be concerned here with Gaussian white additive noise.

5.3.1 Data Error Rate Requirements

It is the responsibility of the telemetry system designer to collaborate with Project personnel to establish meaningful specifications of data error rate requirements. These specifications must be established early in the planning phase of the mission so that appropriate design decisions can be made.

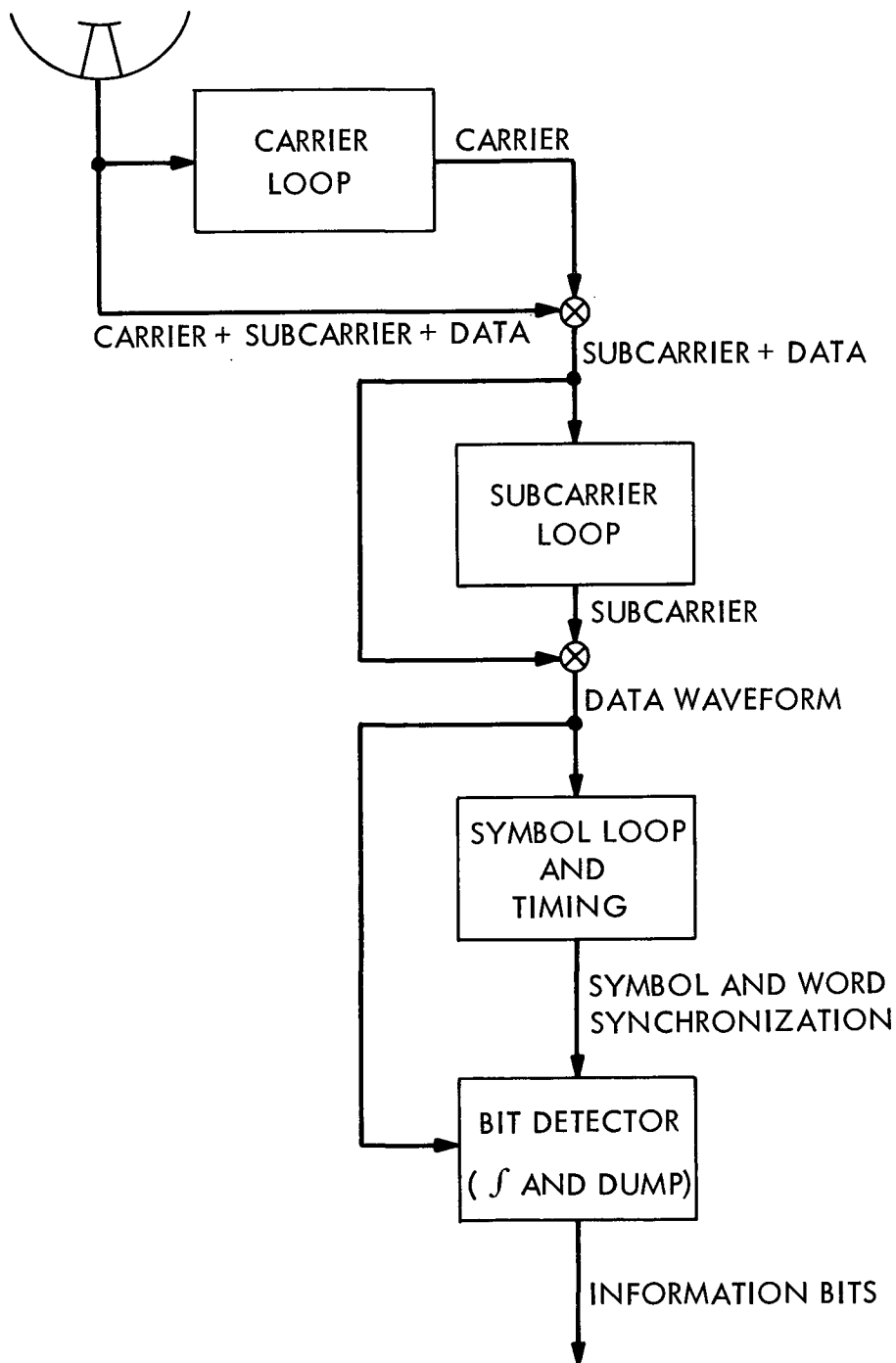


Fig. 5-4. Schematic representation of the demodulation process

There are generally two types of data generated by a deep space vehicle for telemetering to Earth. First, there is data which is inherently digital in nature. Examples of this data are pseudo-noise (PN) sequences used for synchronization of data, status bits, and counting. The quality of this kind of data is described in terms of an average bit error rate.

The second type of data is obtained from instruments or sensors, and is inherently analog. This data may have some noise associated with it generated by the measurement process (cf. ref. 5-3), or it may be essentially clean data. In any event, analog data must be converted to binary digits by some sort of analog-to-digital (A/D) converter and processed for transmission by the telemetry system. The bit stream accepted by the telemetry modulation subsystem will contain bits which stand by themselves, and groups of bits which are data words representing some quantization of analog information.

Data word quality may be specified by their user in terms of either average word error rate, or an average bit error rate, depending upon the use of the data and the convenience of the Project.

The quality of the communication channel is not always identical to the quality of the data provided to the user. The measure of the quality of the received bits depends on the type of channel (whether Gaussian or otherwise) on the type of coding used, and on the method of detection used. Typically, the performance of the telemetry system is measured by the bit error rate at the output of the symbol synchronizer if the data bits are uncoded, or by the word error rate of block coded words at the output of the BDA, or by bit error rates and erasure rates at the output of the DDA for convolutionally encoded, sequentially decoded data. For a purely Gaussian channel, the symbol errors out of the SSA will be independent of each other. If the channel is not Gaussian (for example, if it contains multipath effects), the symbol errors may come in bursts.

Since symbol synchronization is obtained by estimating transitions from the output of the SDA, a low transition density of the transmitted symbols will cause symbol errors to be produced in bursts.

Although the symbol errors may be independent of each other, the detection of block coded words is made on a word basis, hence word errors rather than bit errors are independent.

When the channel is convolutionally encoded and sequentially decoded, two types of errors are made: bit errors and buffer overflows, or erasures. Erasures may be stored on tape so that the received symbols are not lost. However, the time required to decode some erasures may be prohibitively expensive, so that some amount of data will not be provided to the user. When using sequential decoding, the erasure rate must be specified by the project and ground rules for post-mission decoding of the erasures should be established.

Viterbi decoding of data does not produce erasures, since a decision as to the state of every bit is made. Bit errors occur, but they are grouped in bursts of one to two constraint lengths of the code. Also, the performance of the Viterbi algorithm is known to be sensitive to bursts of errors in the incoming symbol stream so that Viterbi decoding would not be appropriate for slow fading multipath channels (fading over many bits).

The elements of block and convolutional encoding and of block, sequential, and Viterbi decoding will be discussed below.

The average error rate measured at the DSN station is not always the same as the error rate specified by the data users. In addition to errors measured at the stations, the ground handling of the data (e.g., transmitting data from the station to the project, reformatting, and processing of the data for presentation to the user) will introduce errors due to occasional loss of frame sync, and occasional data outages.

Typically, errors introduced by the ground system will be of the burst error variety. Thus, specification of an average bit or word error rate to be delivered to the users will insufficiently characterize the ground link. For example, data delivered through the GCF may be essentially error-free for long periods of time and then degrade to unusable error rates for periods (see reference 5-28).

The prime mode of correcting errors made in the GCF portion of the ground link is post-pass playback of the magnetic tape records made at the station. If sufficient playback time is not available to fulfill data quality and quantity requirements, the records can be transported to the Project for post processing. In the case of GCF data outages, near real-time correction may be possible if two DSS's are tracking the vehicle, by bringing up the second station. Thus, it is desirable to plan the mission trajectory to obtain tracking

redundancy at critical periods, if sufficient DSS schedule flexibility is available.

Errors made in Project data processing can be corrected in near-real-time by recalling the affected data from the GCF computer storage devices, located at the JPL terminal of the GCF. The effect of ground or space link errors in the frame sync PN code can be reduced by software adjustment of the allowable number of errors in the code. However, the designer must investigate the likelihood of PN-like sequences in the data and minimize the possibility of false frame sync by proper design of telemetry word time sequence.

While the telemetry system designer is not responsible for ground data handling, he must be aware of the problems so that he can accurately relate the error rates measured at the station to the error rates in the data returned to the user.

5.3.2 Uncoded Channel Performance

In this section, the performance of the uncoded Gaussian channel with white additive noise, will be discussed. The performance of a receiver for a binary digital signal depends on the phase coherence of the receiver and the type of modulation employed at the transmitter. Following Viterbi (Rev. 5-2), a transmitted signal may be written as

$$s_j(t) = C_j(t) \sin \left\{ \omega_o t + \theta_j(t) + \phi \right\} \quad \begin{matrix} 0 < t < T_s \\ j = 0, 1 \end{matrix} \quad (5.3-1)$$

where

T_s = Symbol period

The modulation scheme determines the form of $\theta_j(t)$ and $C_j(t)$ whereas the ability of the receiver to track ϕ is a measure of its coherence.

The receiver which operates on a signal with additive noise sees the waveform.

$$y(t) = s_j(t) + n(t)$$

For a receiver of this type, the optimum detector is a matched filter, or correlation detector. It is this type of detector which is employed by the MMTS to detect transmitted symbols. (For a brief discussion of matched

filter detection, see ref. 5-29; a more general tutorial is contained in ref. 5-30.) The correlation detector can be modeled as in Figure 5-5. The detector model is predicated on the assumption that each transmitted symbol is equally likely, and that each symbol has equal energy.

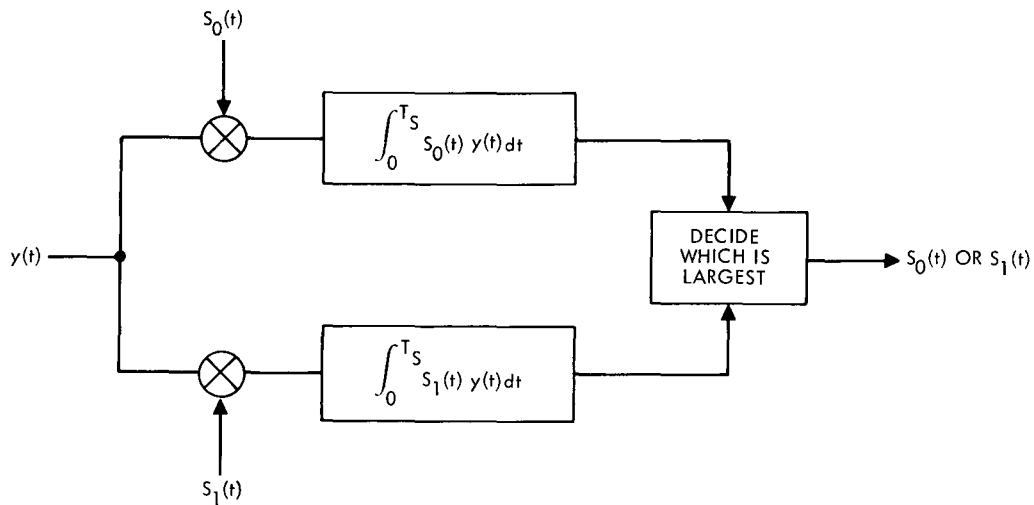


Fig. 5-5. Correlation detector

The energy of each symbol out of the detector is

$$\bar{\epsilon} = \int_0^{T_s} S_0^2(t) dt = \int_0^{T_s} S_1^2(t) dt \quad (5.3-2)$$

where

$\bar{\epsilon}$ = Energy per symbol

A measure of the distinguishability of the two signals is the normalized integral inner product, ρ .

$$\rho \triangleq \frac{1}{\bar{\epsilon}} \int_0^{T_s} S_0(t) S_1(t) dt \quad (5.3-3)$$

What follows assumes that the receiver obtains perfect knowledge of the transmitted phase reference, ϕ , and of the transmitted symbol transition time. Then, as shown in reference 5-2, the probability of symbol error,

P_E , is

$$P_E = 1 - \int_{-\infty}^{\sqrt{\frac{\xi(1-\rho)}{N_o}}} \frac{\exp(-y^2/2)}{\sqrt{2\pi}} dy = \frac{1}{2} \operatorname{erfc} \sqrt{\frac{\xi(1-\rho)}{2N_o}} \quad (5.3-4)$$

where

$$\operatorname{erfc}(x) = 1 - \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$$

If the transmitted symbols S_o and S_1 are 0 and 1, respectively, they are orthogonal ($\rho = 0$).

If they are 1 and -1, they are antipodal ($\rho = -1$). Antipodal signals are ordinarily transmitted by deep space vehicles to prevent difficulties in zero detection in the presence of noise.

If the binary symbols have a signal power of S watts and last for T_B seconds, $\xi = ST_B$. Thus, the error probability for uncoded symbols transmitted over a Gaussian channel and detected in white additive noise is

$$P_E = \frac{1}{2} \operatorname{erfc} \sqrt{\frac{ST_B}{N_o}} \quad (5.3-5)$$

It is this characterization of the error rate of the uncoded channel which is used to specify a design point value of ST_B/N_o for the uncoded telemetry system. Figure 5-6, for $k = 1$, is the bit error performance curve of an uncoded channel with perfect carrier reference and perfect symbol sync.

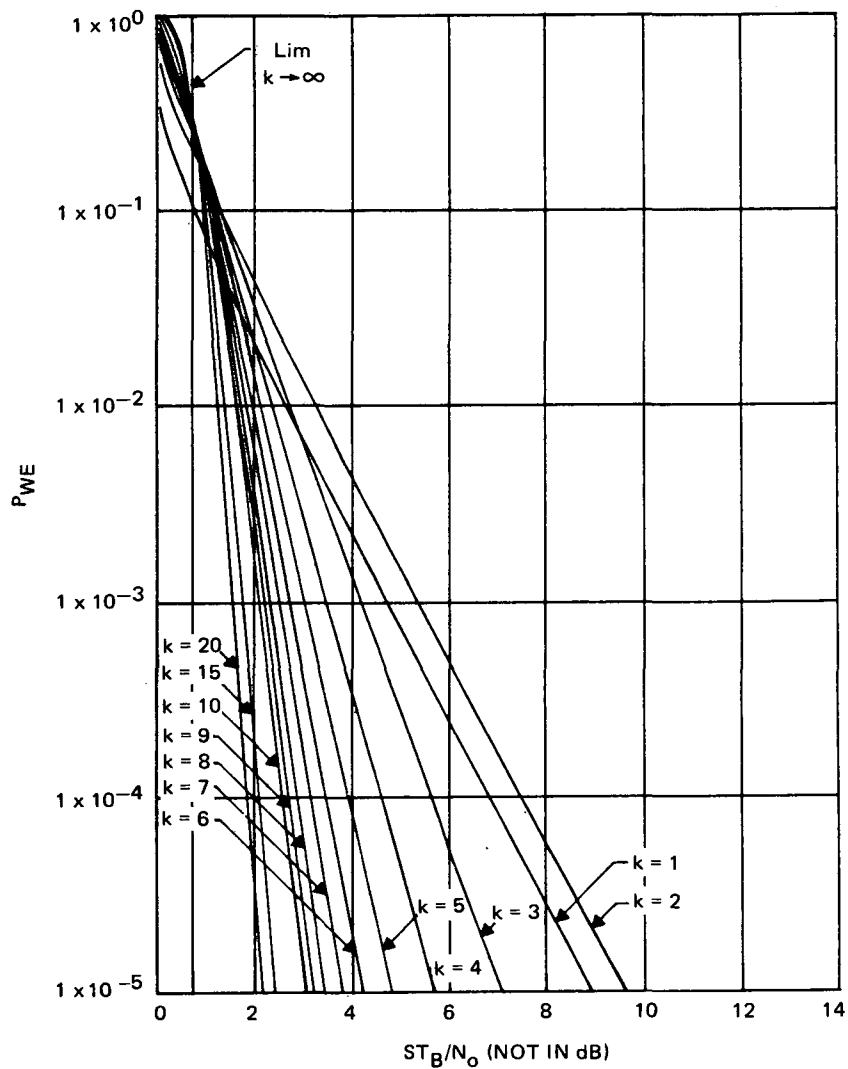


Fig. 5-6. Word error probability versus ST_B/N_o for a $(2^{k-1}, k)$ biorthogonal block code

(Ref. 5-4, A. J. Viterbi, 'Phase-Coherent Communication over the Continuous Gaussian Channel' in S. W. Golomb, Ed., DIGITAL COMMUNICATIONS WITH SPACE APPLICATIONS, Copyright 1964. Reprinted by permission of Prentice-Hall, Inc., Englewood Cliffs, New Jersey.)

In the Gaussian channel we are discussing, symbol errors are independent thus, for an uncoded channel, the probability of word error in a word length of k bits is:

$$P_{WE} = 1 - (1 - P_E)^k \quad (5.3-6)$$

Table 5-1 tabulates bit error probability versus ST_B/N_O for an uncoded channel (from ref. 5-4).

5.3.3 Channel Coding

A classic result in information theory is that the error probability can be reduced if the information bandwidth is increased. Coding is the process of mapping k data bits into n code symbols to expand the transmission bandwidth in a particular way. Decoding is the inverse process performed by mapping the detected n symbol estimates into k data bits. If an efficient code is selected and a good decoding scheme used, the k data bits will have a lower average bit error probability for the same ST_B/N_O than if they were transmitted uncoded.

The MMTS is equipped to decode both biorthogonal block code and convolutional codes. The Block Decoder Assembly (BDA) accepts code symbols from the SSA and decodes either (16, 5) or (32, 6) biorthogonal code words. The Data Decoder Assembly (DDA) is a small computer which is programmed to decode a rate 1/2, constraint length 32, convolutional code using a sequential decoding algorithm. There are plans to develop a Viterbi decoding algorithm for the DDA which would decode short constraint length convolutional codes.

5.3.3.1 Block Codes. When a block of data bits taken k at a time are mapped into a sequence of n symbols taken from a finite set of possible symbol sequences, the process is known as block coding and the code is called an (n, k) block code. Each of the 2^k possible data words is uniquely mapped into one of 2^n possible code words. If each bit in the data word is independent of the other bits, and is equally likely to be a "1" or "0", then the possible code

Table 5-1. Bit error probability versus ST_B/N_o for an uncoded channel*

ST_B/N_o		P_E	ST_B/N_o		P_E
Ratio	dB		Ratio	dB	
.080	-10.969	.34458	3.125	4.948	.00621
.125	-9.031	.30854	3.380	5.289	.00466
.180	-7.447	.27425	3.645	5.617	.00347
.245	-6.108	.24196	3.920	5.933	.00256
.320	-4.948	.21185	4.205	6.238	.00187
.405	-3.925	.18406	4.500	6.532	.00135
.500	-3.010	.15865	4.805	6.817	.0009689
.605	-2.182	.13566	5.120	7.093	.0006882
.720	-1.427	.11507	5.445	7.360	.0004843
.845	-.731	.09680	5.780	7.619	.0003377
.980	-.088	.08056	6.125	7.871	.0002333
1.125	0.512	.06681	6.480	8.116	.0001597
1.280	1.072	.05480	6.845	8.354	.0001082
1.445	1.599	.04457	7.220	8.585	.0000727
1.620	2.095	.03593	7.605	8.811	.0000484
1.805	2.565	.02872	8.405	9.245	.0000209
2.000	3.010	.02275	7.245	9.659	.0000087
2.205	3.434	.01787	10.125	10.054	.0000036
2.420	3.838	.01391	11.045	10.432	.0000015
2.645	4.224	.01073	12.005	10.794	.0000006
2.880	4.594	.00820			

*Source: Ref. 5-4, A. J. Viterbi, 'Phase-Coherent Communication over the Continuous Gaussian Channel' in S. W. Golomb, Ed., DIGITAL COMMUNICATIONS WITH SPACE APPLICATIONS, Copyright 1964. Reprinted by permission of Prentice-Hall, Inc., Englewood Cliffs, New Jersey.

word signals $s_n(t)$ which are generated are all equally probable and therefore contain the same energy ξ .

The detector for block-coded words for coherent reception in white noise can be modeled as a device consisting of 2^k correlators whose outputs are sampled at the word time, T_w . The correlator outputs are compared, and the data word corresponding to the largest correlator output is chosen as the detected word. (The physical mechanization of the block decoder (BDA) is slightly different than this model.)

The performance of the block code generated depends on the distinguishability of the code words. A measure of this distinguishability is the set of normalized inner products.

$$\rho_{rn} = \rho_{nr} = \frac{\int_0^{T_w} S_n(t) S_r(t) dt}{\xi} \quad (5.3-7)$$

Orthogonal codes are a class of block codes for which $n = 2^k$, $\rho_{nr} = 0$ when $n \neq r$, and $\rho_{nr} = 1$ when $n = r$.

For this class of codes, the quantity ρ_{nr} may be expressed as the identity matrix,

$$\rho_{nr} = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & & \\ \vdots & & \ddots & \\ 0 & & & 1 \end{bmatrix} \quad (5.3-8)$$

Such a code can be represented by a Hadamard matrix. For example, if $k = 1$, and $n = 2$,

$$H_1 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \text{ is the code dictionary for the data words } \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \text{ that is, a "0" data bit becomes the symbol sequence 00, and a "1" data bit becomes 01.} \quad (5.3-9)$$

For $k = 2, n = 4,$

$$H_2 = \begin{bmatrix} H_1 & H_1 \\ H_1 & \overline{H_1} \end{bmatrix} = \begin{bmatrix} 0 & 0 & | & 0 & 0 \\ 0 & 1 & | & 0 & 1 \\ 0 & 0 & | & 1 & 1 \\ 0 & 1 & | & 1 & 0 \end{bmatrix} \text{ is the dictionary for } \begin{bmatrix} 0 & 0 \\ 0 & 1 \\ 1 & 0 \\ 1 & 1 \end{bmatrix} \quad (5.3-10)$$

In general

$$H_{k+1} = \begin{bmatrix} H_k & H_k \\ H_k & \overline{H_k} \end{bmatrix} \quad (5.3-11)$$

The effective bandwidth required by the block code symbols may be reduced by one-half, if instead of an orthogonal code, a biorthogonal code is used. This code results from combining an orthogonal code with its complement. The biorthogonal code is represented as

$$H_{k+1} = \begin{bmatrix} H_k \\ -\overline{H_k} \end{bmatrix} \quad (5.3-12)$$

and $n = 2^{k-1}.$

The performance of biorthogonal codes has been derived in reference 5-4. The probability of a k -bit word error is shown to be

$$P_{WE} = 1 - \int_{-\sqrt{2ST_W/N_o}}^{\infty} \frac{\exp(-V_1^2/2)}{\sqrt{2\pi}} dV_1 \quad (5.3-13)$$

$$\left[\int_{-V_1 - \sqrt{\frac{2ST_W}{N_o}}}^{V_1 + \sqrt{\frac{2ST_W}{N_o}}} \frac{\exp(-V^2/2)}{\sqrt{2\pi}} dV \right]^{2^{k-1}-1} dV_1$$

where

$$T_W = kT_B$$

assuming that the channel is Gaussian with white additive noise, and that a perfect carrier phase reference and a perfect word timing reference are available at the receiver.

Equation (5.3-13) is used to establish the design point ST_B/N_o necessary to achieve a required P_{WE} . Figure 5-6 (from ref. 5-4) compares P_{WE} versus ST_B/N_o for biorthogonal codes with various choices of k . Figures 5-7 and 5-8 compare uncoded and biorthogonal coded word error rates for the same number of information bits, k , per word.

The information bit error rate of a biorthogonal code is found to be

$$P_{BE} = \frac{1}{2} P_{WE} + \frac{1}{2} P_1 \quad (5.3-14)$$

where

$$P_1 = \int_{-\infty}^{-\sqrt{2ST_W/N_o}} \frac{\exp(-V^2/2)}{\sqrt{2\pi}} \left[\operatorname{erf} \left(\frac{V}{\sqrt{2}} + \sqrt{\frac{ST_W}{N_o}} \right) \right]^{2^{k-1}-1} dV$$

for $k \geq 5$, the bit and word error rates can be approximately related by

$$P_{BE} = \left(\frac{2^{k-1}-1}{2^k-1} \right) P_{WE} \quad (5.3-15)$$

Figure 5-9 plots P_{BE} versus ST_B/N_o for biorthogonal block codes. Tables 5-2 and 5-3 give bit and word error rates versus ST_B/N_o for (16, 5) and (32, 6) biorthogonal codes.

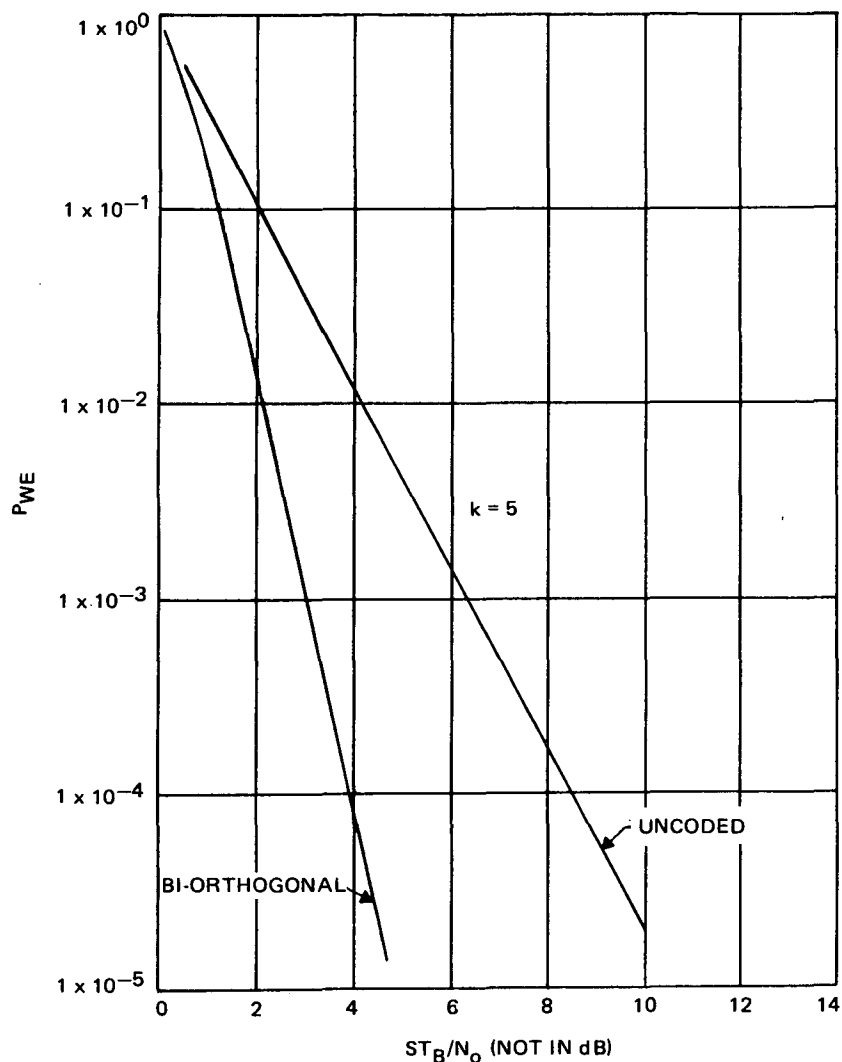


Fig. 5-7. Word error probability versus ST_B/N_0 for uncoded and biorthogonally block coded words of information bit length 5

(Ref. 5-4, A. J. Viterbi, 'Phase-Coherent Communication over the Continuous Gaussian Channel' in S. W. Golomb, Ed., DIGITAL COMMUNICATIONS WITH SPACE APPLICATIONS, Copyright 1964. Reprinted by permission of Prentice-Hall, Inc., Englewood Cliffs, New Jersey.)

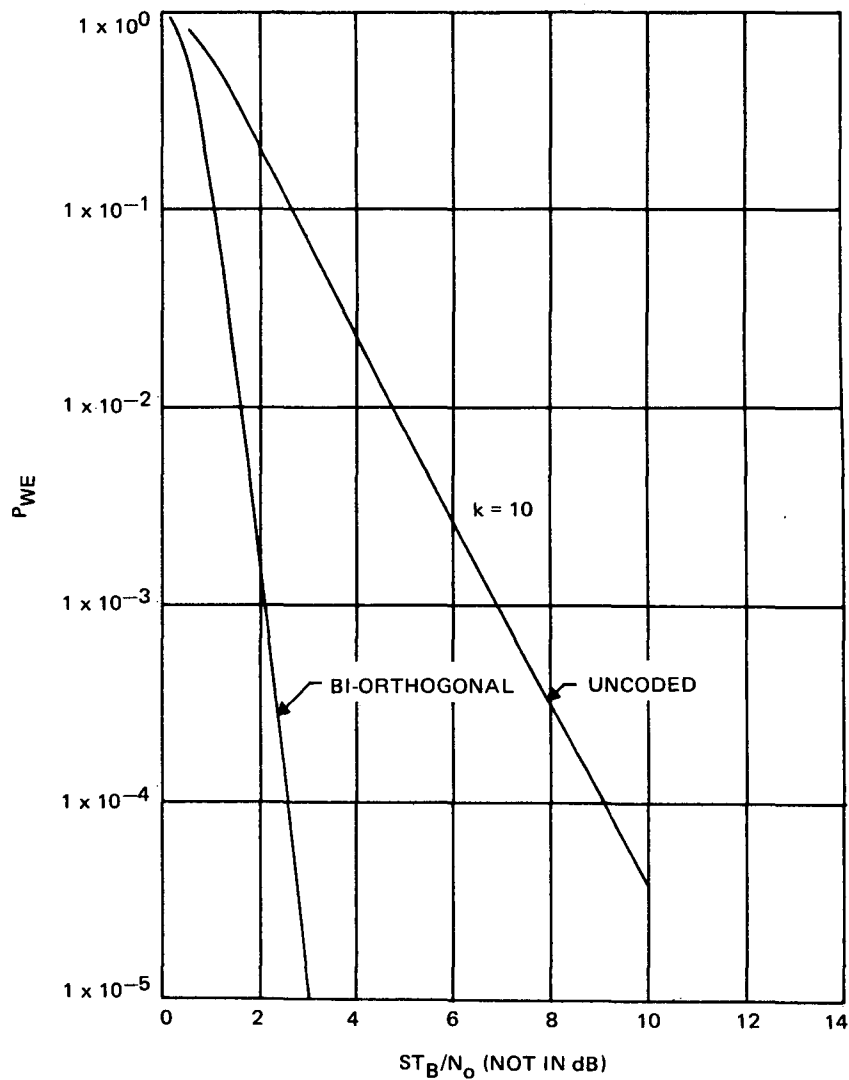


Fig. 5-8. Word error probability versus ST_B/N_0 for uncoded and biorthogonal block coded words of information bit length 10

(Ref. 5-4, A. J. Viterbi, 'Phase-Coherent Communication over the Continuous Gaussian Channel' in S. W. Golomb, Ed., **DIGITAL COMMUNICATIONS WITH SPACE APPLICATIONS**, Copyright 1964. Reprinted by permission of Prentice-Hall, Inc., Englewood Cliffs, New Jersey.)

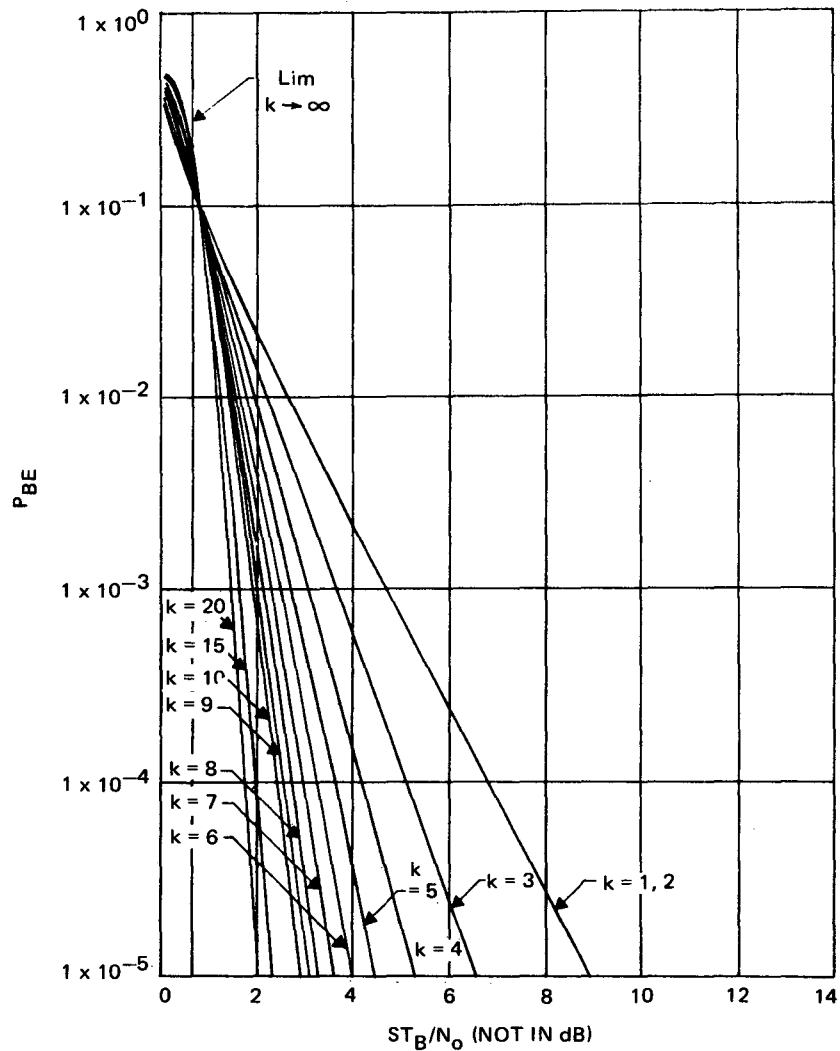


Fig. 5-9. Information bit error versus ST_B/N_0
for $(2^k-1, k)$ biorthogonal block codes

(Ref. 5-4, A. J. Viterbi, 'Phase-Coherent Communication over the Continuous Gaussian Channel' in S. W. Golomb, Ed., DIGITAL COMMUNICATIONS WITH SPACE APPLICATIONS, Copyright 1964. Reprinted by permission of Prentice-Hall, Inc., Englewood Cliffs, New Jersey.)

Table 5-2. Word and information bit error probabilities for a
(16, 5) biorthogonal block code versus ST_B/N_o *

ST_B/N_o		P_{WE}	P_{BE}	ST_B/N_o		P_{WE}	P_{BE}
Ratio	dB			Ratio	dB		
.080	-10.969	.85116	.42736	1.445	1.599	.05913	.02957
.125	- 9.031	.79879	.40033	1.620	2.095	.03947	.01973
.180	- 7.447	.73687	.36891	1.805	2.565	.02554	.01277
.245	- 6.108	.66666	.33356	2.205	3.434	.00976	.00488
.320	- 4.948	.59029	.29525	2.645	4.224	.00330	.0016516
.405	- 3.925	.43065	.25532	3.125	4.948	.0009940	.0004970
.500	- 3.010	.43065	.21535	3.645	5.617	.0002666	.0001333
.605	- 2.182	.35375	.17688	4.205	6.238	.0000640	.0000320
.720	- 1.427	.28265	.14133	4.805	6.817	.0000139	.0000069
.845	- .731	.21945	.10972	5.445	7.360	.0000028	.0000014
.980	- .088	.16545	.08272	6.125	7.871	.0000006	.0000003
1.125	0.512	.12105	.06053	6.845	8.354	.0000002	.0000001
1.280	1.072	.08591	.04295	7.605	8.811	.0000001	.0000000

*Source: Ref. 5-4, A. J. Viterbi, 'Phase-Coherent Communication over the Continuous Gaussian Channel' in S. W. Golomb, Ed., DIGITAL COMMUNICATIONS WITH SPACE APPLICATIONS, Copyright 1964. Reprinted by permission of Prentice-Hall, Inc., Englewood Cliffs, New Jersey.

Table 5-3. Word and information bit error probabilities for a (32, 6) biorthogonal block code versus ST_B/N_o^*

ST_B/N_o		P_{WE}	P_{BE}	ST_B/N_o		P_{WE}	P_{BE}
Ratio	dB			Ratio	dB		
.080	-10.969	.89275	.44690	1.280	1.072	.07518	.03759
.125	- 9.031	.84480	.42284	1.620	2.095	.03054	.01527
.180	- 7.447	.78465	.39243	2.000	3.010	.01063	.00531
.245	- 6.108	.71300	.35654	2.420	3.838	.00318	.00159
.320	- 4.948	.63192	.31597	2.880	4.594	.0008199	.0004100
.405	- 3.925	.54474	.27237	3.380	5.289	.0001831	.0000916
.500	- 3.010	.45560	.22780	3.920	5.933	.0000357	.0000179
.605	- 2.182	.36893	.18447	4.500	6.532	.0000062	.0000031
.720	- 1.427	.28873	.14436	5.120	7.093	.0000010	.0000005
.845	- .731	.21807	.10903	5.780	7.619	.0000002	.0000001
.980	- .088	.15878	.07939	6.480	8.116	.0000001	.0000000
1.125	0.512	.11136	.05568				

*Source: Ref. 5-4, A. J. Viterbi, 'Phase-Coherent Communication over the Continuous Gaussian Channel' in S. W. Golomb, Ed., DIGITAL COMMUNICATIONS WITH SPACE APPLICATIONS, Copyright 1964. Reprinted by permission of Prentice-Hall, Inc., Englewood Cliffs, New Jersey.

5.3.3.1.1 Mechanization of the Block Encoder. On board the spacecraft, a unit known as a block coder is provided to accept data bits in blocks and convert them into code symbols. An n-symbol block coded word, $\underline{S}_i$, is generated from a k-bit data word $\underline{D}_i$ in accordance with the matrix operation.

$$\underline{S}_i = \underline{G} \oplus \underline{M} \oplus \underline{D}_i \quad (5.3-16)$$

Where $\underline{G}$ and $\underline{M}$ are matrices which determine the code, and $\oplus$ indicates matrix multiplication followed by modulo 2 reduction of the resulting matrix elements.

The BDA is designed to decode the (n,k) biorthogonal codes (16, 5) and (32, 6). The matrices $\underline{G}$ and $\underline{M}$ are shown in Figures 5-10 and 5-11. Synchronization of the code words in the decoder is facilitated by half-adding to the $\underline{S}_i$ code symbols a PN sequence called a Comma Free Vector, $\underline{C}$. The sequence is 2^k symbols long, so that no transmitted sequence of n symbols overlapping two successive code words can be a member of the dictionary of possible code words. The transmitted symbols are thus $\underline{S}_i \oplus \underline{C}$.

$$\underline{M} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{bmatrix} \left. \vphantom{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{bmatrix}} \right\} 5$$

$$\underline{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} \left. \vphantom{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}} \right\} 16$$

Fig. 5-10. $\underline{G}$ and $\underline{M}$ matrices for a (16, 5) biorthogonal block encoder

$$\underline{M} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \left. \vphantom{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}} \right\} 6$$

$$\underline{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

Fig. 5-11. $\underline{G}$ and $\underline{M}$ matrices for a (32,6) biorthogonal block encoder

The implementation of the block coder is shown schematically in Figure 5-12a.

5.3.3.1.2 Decoding Received Symbols. The ground received symbols are demodulated and output through an RC filter in the subcarrier demodulation assembly (SDA), as described above. The partially integrated symbol stream is digitized by an ADC in the symbol synchronizer assembly (SSA). The SSA then applies a correction to the digitized, RC filtered signal, to approximate the output which would be observed from a matched filter (perfect integrator).

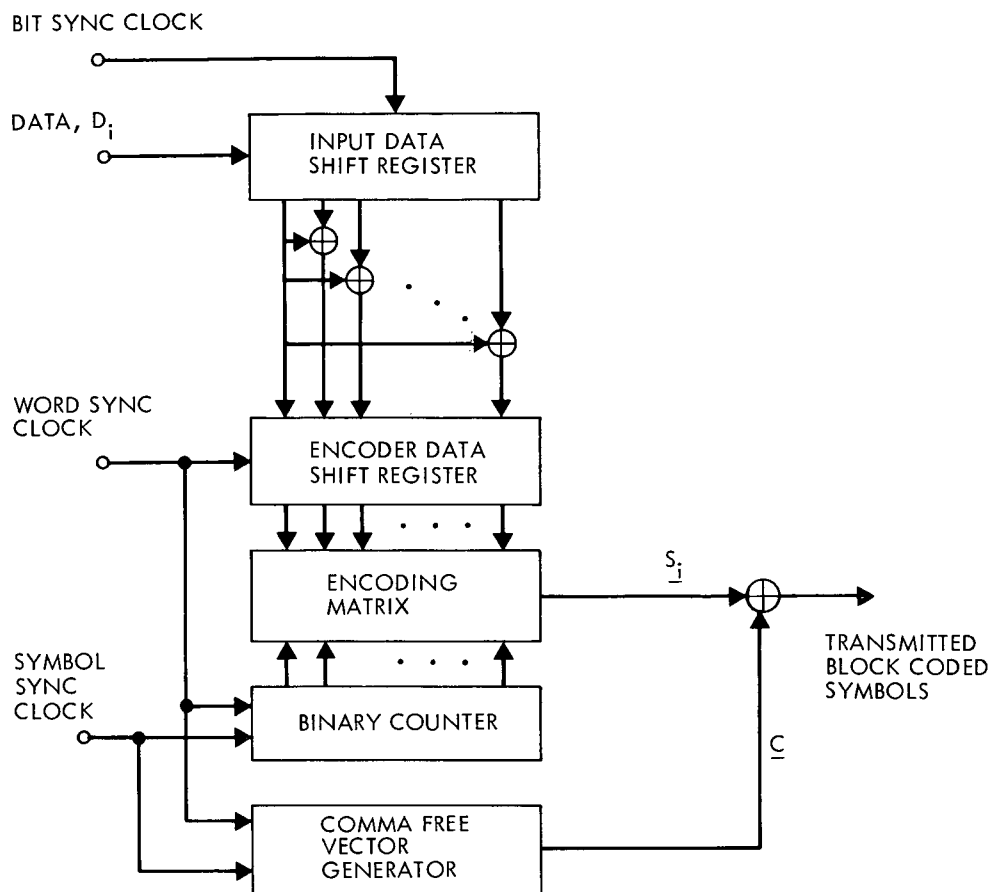


Fig. 5-12a. Encoder block diagram

In order to obtain and maintain symbol sync the SSA calculates the integral from midsymbol to midsymbol across a transition. If this integral is zero, symbol sync has been obtained. Non-zero values generate an error signal which drives the symbol sync loop to the correct lock point. This symbol sync loop is a phase-locked loop with a digital loop filter.

Word sync is derived in the block decoder assembly (BDA). A correlation is performed between an input word length of symbols and each word of the code dictionary, and the ratio of the largest correlation value to the typical correlation value is computed and stored. The BDA then shifts the candidate word sample by one symbol and repeats the process until the number

of shifts equals the number of symbols in a code word. Because of the comma free nature of the input, the above ratio is much higher for in-sync words than for out-of-sync position choices, therefore the BDA shifts the data to that word sample with the highest ratio and announces word sync.

Following word sync acquisition, the BDA decodes the symbol stream to data bits by selecting that bit pattern in its dictionary which corresponds to the highest correlation between the symbols and the stored symbol possibilities. Thus, if k -bit words are used, 2^k correlations must be performed per 2^{k-1} word synchronized symbols. This maximum likelihood detection is performed by 2^k parallel correlators, which, because of the ± 1 waveform, perform correlation by simple addition. The most likely words are then formatted and output for transmission to the users.

This process is depicted in Figure 5-12b. A comprehensive description of the block decoding principles is available in reference 5-5.

5.3.3.2 Convolutional Coding. Another code class coming into favor for deep space communication is convolutional codes. At present, the theory for convolutional coding must be supplemented by trail-and-error code choice and computer simulated code performance investigations, in contrast to the closed-form solutions for block codes (cf. references 5-9, 5-10, 5-11). These methods, however, have produced performance uniformly better than that of block coding for a given symbol-rate to data-rate ratio. In addition, the encoder implementation is simpler and thus less costly and more reliable for spacecraft operation than that of a block encoder.

Binary convolutional codes may be generated in the manner shown in Figure 5-13. The binary input data is shifted bit by bit into a K -stage shift register. With every shift, the commutator samples v output lines from adder 1 through adder v and forms a binary symbol output of the serial samplings called a code branch. Thus, if the input bit rate is $1/T_B$, the output symbol rate, $1/T_s$, is equal to $1/bT_B$. Such an encoder generates a "constraint length K , rate $1/v$ (bits/symbol) convolutional code." K is called the constraint length because each input bit affects, or constrains, the subsequent Kv output symbols derived from K input bit shifts. This results in a code which is sensitive to burst errors.

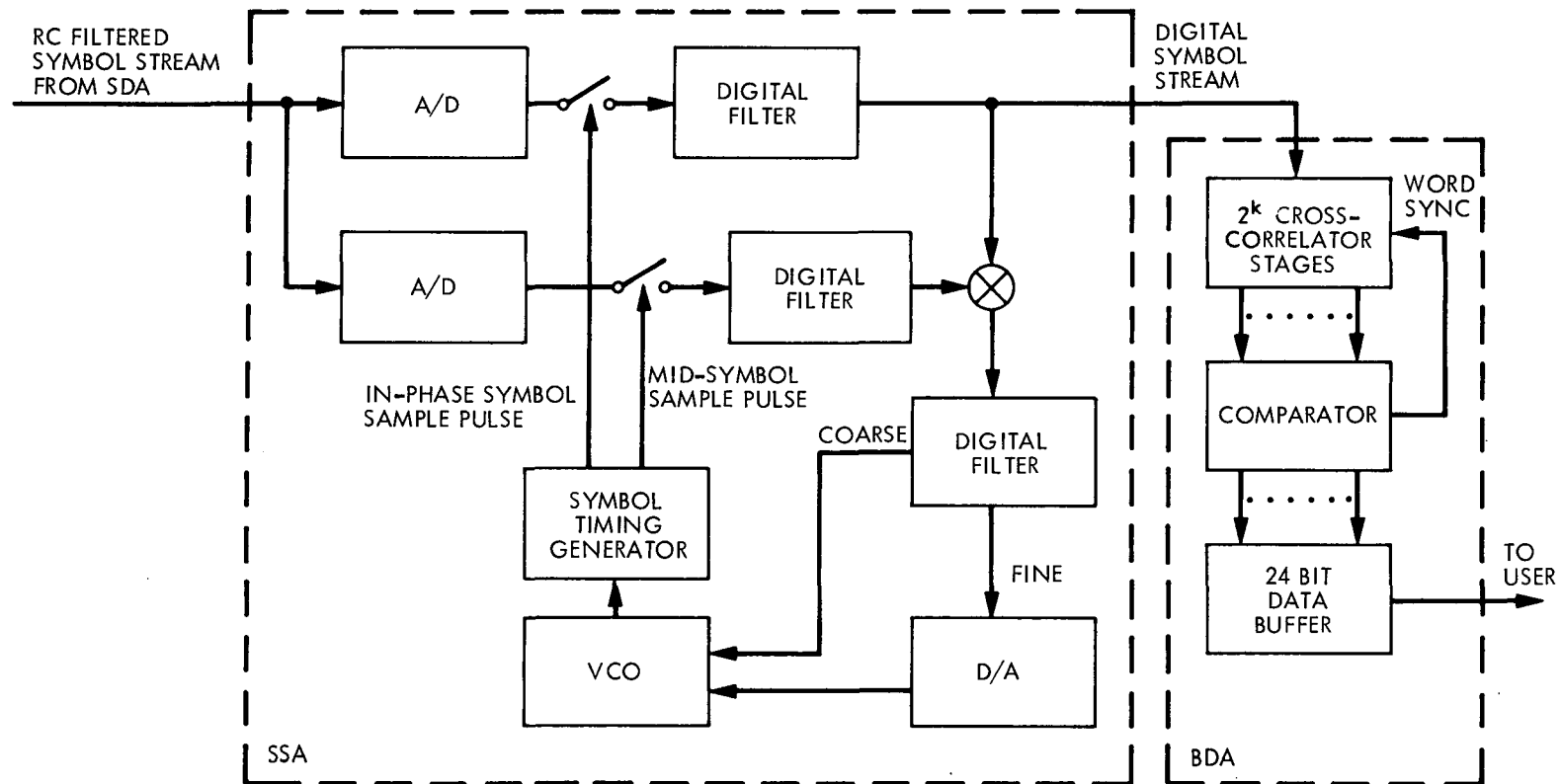


Fig. 5-12b. Conceptual model of block decoding

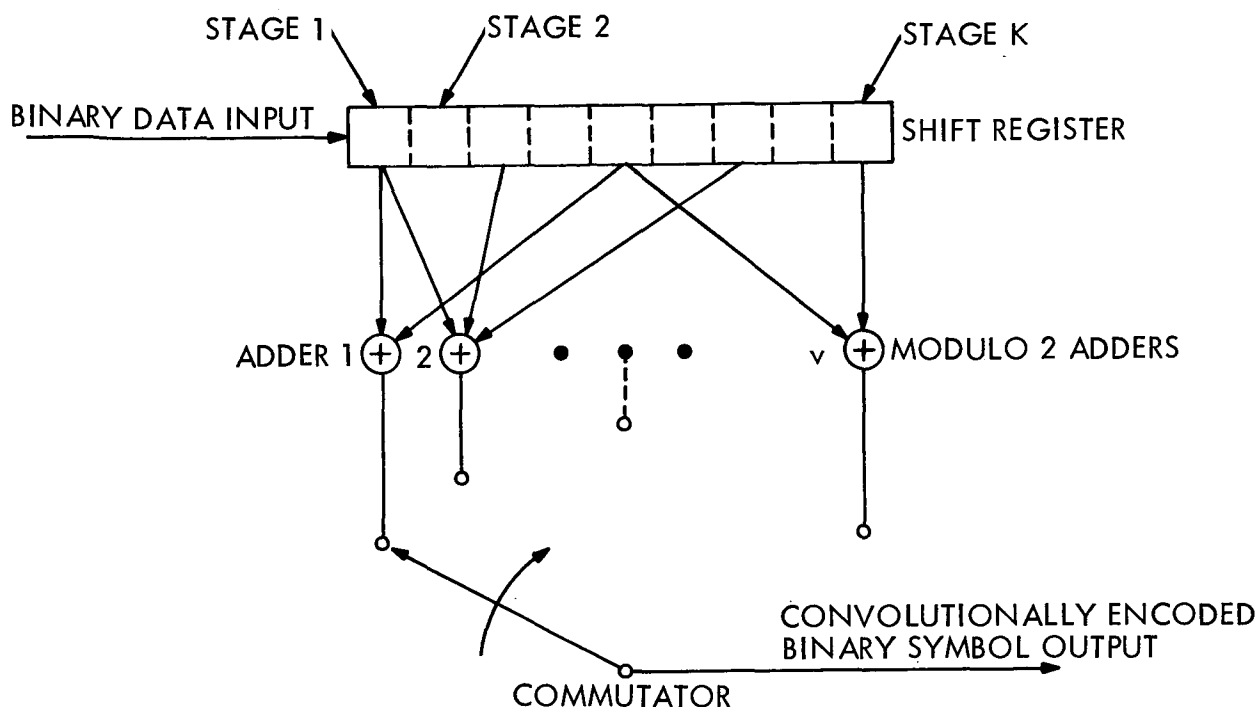


Figure 5-13. Length K, rate $1/v$ convolutional encoder
(Source: Reference 5-12 with permission)

In general, the binary input data is entered b bits per commutator cycle into a bK -stage shift register. Thus, if the input bit rate is $1/T_B$, the shift register accepts a new input every bT_B seconds, and the output symbol rate is equal to v/bT_B . This encoder generates a "constraint length K , rate b/v (bits/symbol), convolutional code."

The particular code sent depends on the manner in which the adders are connected to the shift register. These connections are denoted by a set of vectors

$$\bar{g}_\ell = (g_{\ell 1}, g_{\ell 2}, \dots, g_{\ell v}) \quad \ell = 1, 2, \dots, K \quad (5.3-17)$$

where $g_{lj} = 1$ denotes a connection between the l th stage of the shift register and the j th adder, and $g_{lj} = 0$ denotes the absence of a connection. The complete set of $\bar{g}_l$ and the parameter b define the code. Figure 5-14 demonstrates the meaning of $\bar{g}_l$.

If the data bits are sent as part of the output, the code is termed "systematic," otherwise, it is "nonsystematic." Thus, a systematic code is one for which

$$g_{lj} = \delta_{lj}; l = 1, 2, \dots, K, j = 1, 2, \dots, b \text{ where}$$

δ_{lj} is the Kronecker delta

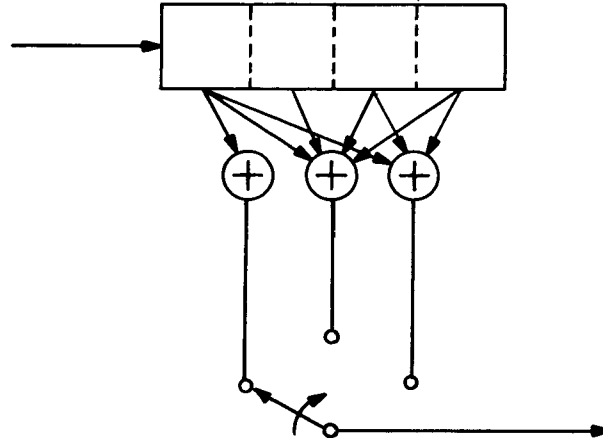
$$\delta_{lj} = \begin{cases} 1, l = j \\ 0, l \neq j \end{cases}$$

A code is also termed systematic if the data appears in other than the 1 through b th stages, however, restricting the data to these stages is merely a relabeling process for stage numbers in the general case.

It can be shown (reference 5-15) that for K large the performance of a systematic code of constraint length K is approximately equal to the performance of a nonsystematic code of constraint length $K(1 - b/v)$. Thus, systematic codes have significantly poorer performance than nonsystematic for equal encoder and decoder complexity.

However, systematic codes have the advantage of the quick-look property. That is, since the data is present explicitly in the symbol stream, it can be examined without decoding (with the symbol error rate). Massey and Costello (reference 5-16) have demonstrated that for $b = 1$, nonsystematic codes can be developed for which the data can be recovered explicitly by adding the symbols produced by each modulo 2 adder of the encoder. This gives a nonsystematic code with the quick-look property for which the data error rate is approximately v times the symbol error rate. Such a code will give a data bit error only if there is an odd number of errors in the length v symbol branch. Thus, if P_s is the probability of symbol error, the probability of error in the quick-look data bit is,

The $K = 4$, $v = 3$, $b = 1$, encoder shown



has the connection vector description

$$\bar{g}_1 = (1, 1, 1)$$

$$\bar{g}_2 = (0, 1, 0)$$

$$\bar{g}_3 = (0, 1, 1)$$

$$\bar{g}_4 = (0, 1, 1)$$

or, in matrix form

$$\{\bar{g}_{lj}\} = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix}$$

Figure 5-14. Example for connection vector notation.
(Source: Reference 5-12, with permission)

$$P_{QL} = \sum_{k=0}^B \frac{v! P_s^{2k+1} (1-P_s)^{v-(2k+1)}}{[v - (2k+1)]! (2k+1)!} \quad (5.3-18)$$

where

$$B = \left\lfloor \frac{v-1}{2} \right\rfloor$$

$\lfloor x \rfloor$ indicates $\lfloor x \rfloor =$ the greatest integer $\leq x$.

and we have assumed branch synchronization, that is, the decoder knows which adder generated each symbol.

In order to send a block of L data bits, the shift register is initialized to the all-zeros state. The bits are then shifted in, b by b until all L have been shifted out of the register and it is again filled with zeros. Thus, L input bits give $(L + bK) v/b$ output symbols including v initial output zeros, where Kv is the minimum "tail" for a data symbol block. The data rate then is

$$R_N = \frac{b}{v} \left(1 + \frac{bK}{L} \right)^{-1} \text{ bits per symbol} \quad (5.3-19)$$

or

$$R_N = \frac{b}{v} \text{ for } L \gg bK, \text{ the typical case.}$$

This section has defined the terms referred to in specifying a convolutional code. Next, we shall discuss the problem of decoding convolutionally encoded data.

5.3.3.2.1 Decoding Convolutional Codes. It can be shown (reference 5-12), that the probability of error for optimum decoding is bounded by a function times 2^{-K} . However, a brute force maximum likelihood optimum decoder requires 2^L computations to decode each block of length L bits and tail K bits. Such a decoder would severely limit bit rates with even extremely fast computation capability. Thus, methods have been found which permit decoding at reasonable bit rates with excellent, though less than optimum performance.

Two of the most widely known and analyzed schemes for non-optimum decoding are sequential decoding by the Fano algorithm and a modified maximum likelihood (Viterbi) decoding algorithm. The Fano algorithm has been demonstrated operationally with a software system for Pioneer IX. Very fast Fano and Viterbi machines now exist and may soon provide a DSIF operational system. Reference 5-8 tabulates the available parameters for DSN-compatible coding schemes using these decoding methods. The following sections give brief descriptions of these methods and the parameters necessary for performance prediction. References are cited for detailed investigation.

5.3.3.2.1.1 Sequential Decoding - The Fano Algorithm. In any decoding scheme, for any type of code, a measure must be found for determining how much better one data hypothesis is than another. This measure is known as the "metric" and depends on the type of code, the method of decoding and the type of transmission channel. The general methods of metric determination are discussed in reference 5-12. This reference also presents a good discussion of sequential decoding and the Fano algorithm.

The brute force optimum decoding process for convolutional codes determines the metric for each of the 2^L possible code paths influenced by each L bits (Lv symbols) and picks the best of these metric values as that connected with the correct data hypothesis for that path. (Note that henceforth we shall assume $b = 1$.)

The approach of sequential decoding is to search the code bit-by-bit (v symbols per bit) and select an initial data hypothesis for the first v symbols encountered whose metric lies above a preset threshold value. This threshold is chosen to lie between the known values of the error-free metric and the nearest incorrect-hypothesis metric. Since each data hypothesis affects Kv symbols, an incorrect assumption will generate a metric which increasingly diverges from the error-free value. When the calculated metric crosses threshold, the previous data bit assumption is ruled invalid and the computer backs up v symbols to attempt another hypothesis. If this change of assumption does not work, the computer retreats another v symbols. The process continues until an acceptable path is found.

For very noisy data, all 2^K branches may be examined; if this occurs without success, the threshold criterion is loosened, and the attempt is repeated. In general, however, many paths need not be scrutinized, and thus a computational saving results. For example, if the first assumption is within threshold, 2^{K-1} other trials may be eliminated, with an exponential saving in number of computations. If an error is not detected within Kv symbols, the error is undetectable by the algorithm, and is passed by the system.

As indicated above, as the signal-to-noise ratio decreases, the required number of computations per v symbols increases. If the decoding device has a storage capacity, ΓQ computer bits, assigned to store the received data, where

$$Q = \text{Receiver output per } v \text{ symbols (metric) quantization in bits}$$

and

$$\Gamma = \text{Number of } Q\text{-length storage words}$$

and if

$$\lambda = vT_s / T_c$$

where

$$T_s = \text{Symbol time}$$

$$vT_s = \text{Time to receive one data bit}$$

and

$$T_c = \text{Computational time required to search through one data}$$

hypothesis (one code branch), then if B , the number of computations on a branch exceeds $\lambda\Gamma$, the storage capacity of the decoder will be exceeded. This occurrence is known as overflow or erasure.

An overflow can occur for $B < \lambda\Gamma$ if the previous computations have filled storage. For a low error-rate system, however, the probability of this situation must be small and thus the probability of overflow is approximately $P(B > \lambda\Gamma)$. (When high error rates are acceptable, other expressions

are valid, cf. reference 5-14). When an overflow occurs, the decoder must resynchronize on the data. This is the reason for finite block size on the transmitted sequence. The decoder discards the remainder of the block and resynchronizes on the tail of K or more zero bits. The probability that all or part of a data block will be lost is approximately $LP(B > \lambda \Gamma)$.

For $P(B > \lambda \Gamma)$ to be small, a value of $\Gamma \gg K$ is required, so that Q , Γ , and λ determine the storage requirements. Each decoded hypothesis requires 1 bit, so that the required storage is approximately $\Gamma(Q + 1)$ bits.

For any sequential decoding scheme, there is a data rate R_o' bits per symbol (also referred to as R_{comp} in the literature) above which $\bar{B}$, the average number of branches searched per bit decoded, is unbounded. Thus, we require that $R_N < R_o'$. R_o' depends on Q , the metric quantization, and the type of transmission channel. In general,

$$R_o' = -\log_2 \sum_{h=1}^Q \left[\sum_{\ell=1}^A p_{\ell} \sqrt{q_{\ell h}} \right]^2 \quad (5.3-20)$$

where

A = Number of signal levels transmitted

p_{ℓ} = The probability of transmitted signal level ℓ

and

$q_{\ell h}$ = Probability of receiving metric h given that signal ℓ was sent.

For an additive white gaussian noise (AWGN) channel with $Q = \infty$, and antipodal signals,

$$R_o \stackrel{\Delta}{=} \lim_{Q \rightarrow \infty} R_o' = 1 - \log_2 (1 + \exp(-ST_B R_N / N_o)) \quad (5.3-21)$$

In the limit of infinite bandwidth expansion, ST_B/N_0 approaches $2 \ln 2 = 1.42$ dB or 3 dB above Shannon's limit. For a quantization $Q = 8$, this value increases by about 0.2 dB, while for $Q = 2$ (i. e., hard decision) about 2 dB is lost (see ref. 5-13). Figure 5-15 plots $R_{N_{\max}}$ versus ST_B/N_0 for $Q = \infty$.

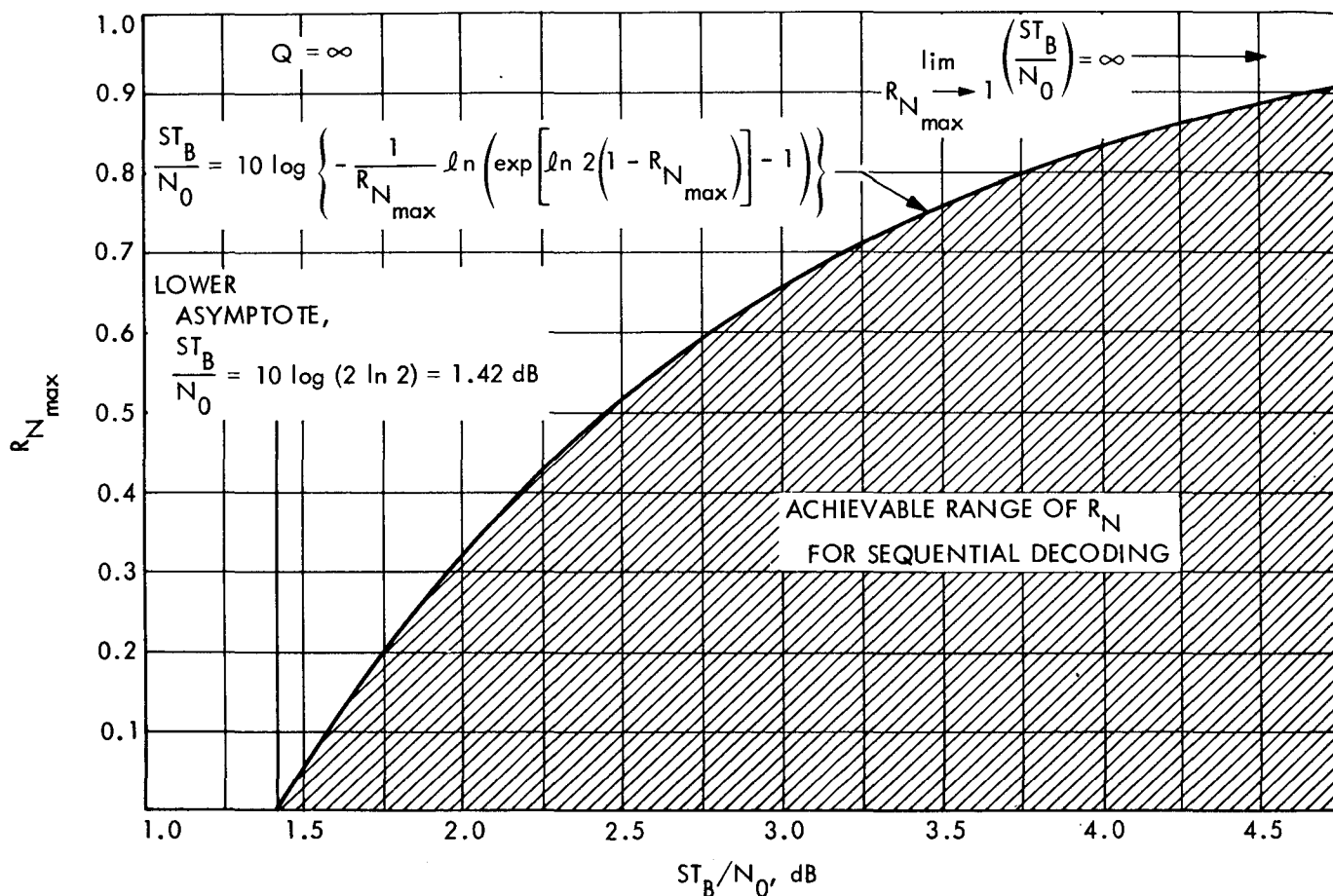


Fig. 5-15. Maximum achievable code rate for sequential decoding with infinite metric quantization versus ST_B/N_0

The Fano sequential decoding algorithm is distinguished by its use of a running threshold. As long as the metric is growing, this algorithm searches a single path and tightens the threshold value. When forced to retreat, the algorithm loosens the threshold until a renewed forward search becomes possible. Reference 5-12 discusses the algorithm in detail.

Simulations by Bluestein and Jordan (as quoted in reference 5-12) have demonstrated some fundamental properties of the Fano algorithm:

- 1) The mean number of branches searched, per branch received, $\overline{B}$ depends primarily on the ratio R_N/R_O' and grows very rapidly for $R_N/R_O' > 0.9$.
 - 2) The probability of undetectable error decreases as 2^{-K} and can easily be made negligible with respect to errors caused by overflow for K 's which are orders of magnitude less than Γ .
 - 3) $P(B > \lambda \Gamma) \approx 3^{-(1 - R_N/R_O')} (\lambda \Gamma)^{-(2.9 - 2 R_N/R_O')}$
- (5.3-22)

for $\lambda \Gamma \gg 1$, $R_N < R_O'$, and $\lambda > \overline{B}$

Thus, for a system with $2^{-K} \ll LP(B > \lambda \Gamma)$ and $Kv \ll L$, the probability of bit error is negligible when compared to the rate of bit discards due to overflow. This rate P_D , is just equal to one-half the probability of block error, since overflow will happen on the average when one-half of the block is decoded.

Thus,

$$P_D = \frac{L}{2} P(B > \lambda \Gamma) \quad (5.3-23)$$

Using Bluestein's and Jordan's empirical formula (5.3-22) for an AWGN channel with $Q \rightarrow \infty$, a little algebra demonstrates that for a probability of bit discard P_D ,

$$ST_{B/N_O} \approx - \frac{v}{\log e}$$

$$\log \left\{ \exp \left[\frac{\log 2}{\log e} \left(1 - \frac{1}{v} \frac{\log 3 + 2 \log \lambda \Gamma}{\log P_D + \log 6 + 2.9 \log \lambda \Gamma - \log L} \right) \right] - 1 \right\} \quad (5.3-24)$$

where ST_B/N_O is expressed as a fraction, not in dB, $\lambda\Gamma \gg 1$, $R_N < R_O$ over the range of ST_B/N_O considered. These bit discards will occur in bursts of average length $L/2$.

Equation 5.3-24 assumes a "good" code, and does not take the place of simulation runs with the actual code design.

It should be noted that overflows can be processed to yield good data by playing back erased blocks into the decoder at a slower rate. The breakpoint for such additional processing will occur where the additional data gained does not justify the additional processing cost.

Figure 5-16 from reference 5-10 shows some simulation results for Fano sequential decoding with $K = 32$ and $R_N = 1/2$. The letters represent

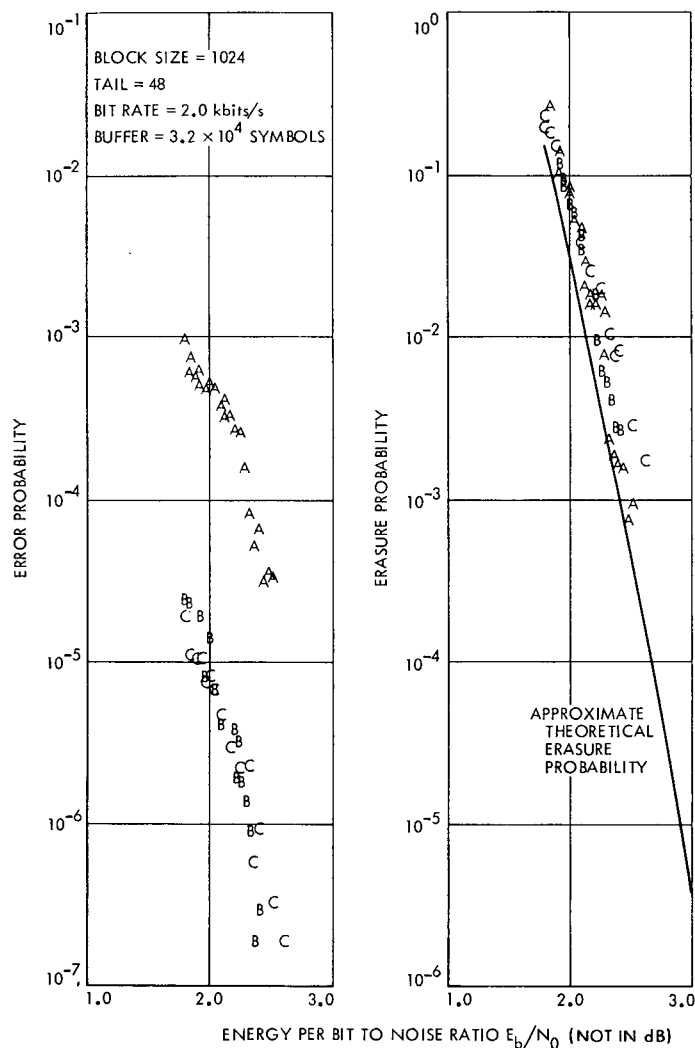


Fig. 5-16. Bit error and block erasure probabilities for a JPL experimental Fano sequential decoder.

three different codes. Code A is a systematic code, code B is a nonsystematic quick-look code and code C is nonsystematic. The curve "Approximate Theoretical Erasure Probability" was obtained from the results of reference 5-14, rather than equation (5.3-24).

Reference 5-32 presents a method by which erasure probabilities can be reduced at the sacrifice of chronologically ordered data. The method is based on the theories of computer time-sharing and can achieve marked improvement at low erasure rates (10^{-2} or less) and slight-to-moderate improvement above that level.

5.3.3.2.1.2 Maximum Likelihood (Viterbi) Decoding. Maximum likelihood, or Viterbi, decoding is a nonsequential decoding method which takes advantage of the fact that a single information bit can affect only K_v symbols (K branches) in the received code. Thus, it proceeds through the received symbols by choosing the most likely path out of the entire ensemble of possible transmitted paths for each group of K branches. For binary data, the algorithm proceeds in the following manner:

- 1) Each branch can only be reached by two paths. Compute the metric for both paths entering each branch and discard the paths with the smaller pairwise metric. Thus for 2^K paths, eliminate one half, leaving 2^{K-1} survivors.
- 2) At the next branch, we again have 2^K paths since each can go two ways. Repeat step 1, multiplying the branch metrics by the stored value of each path metric, and again eliminate one half of the possible paths leaving 2^{K-1} survivors.
- 3) Continue this process indefinitely.

This procedure is the optimum decoding method for convolutional codes if continued through a block into the tail of $K-1$ known data bits. These bits eliminate all but one of the stored 2^{K-1} paths to give the best possible estimate of the data sequence. However, without this tail, the Viterbi decoder in the above form will never make a decision as to the correct path. Therefore, the algorithm is modified to make a data choice after proceeding several constraint lengths past a data bit.

An additional error probability is introduced by this procedure. For a very noisy channel ($E_S/N_O \ll 1$, $E_S = ST_S$) the error due to this truncation is less than the probability of bit error if (reference 5-18)

$$\frac{M}{K} > \begin{cases} \frac{1}{1 - 2R_N/C} & 0 \leq R_N \leq C/4 \\ \frac{1}{2(1 - \sqrt{R_N/C})^2} & C/4 \leq R_N \leq C/2 \\ \frac{1 - R_N/C}{(1 - \sqrt{R_N/C})^2} & C/2 \leq R_N \leq C \end{cases} \quad (5.3-25)$$

where

M/K = Number of constraint lengths to a decision

and

$C = E_S/(N_O \ln 2)$ = channel capacity

For example, for a rate $R_N = C/2$, $M > (5.8) K$; thus we require 2^{K-1} shift registers of $(5.8) K$ length for parallel computation. Figure 5-17 plots this relationship versus ST_B/N_O .

With this modification, Viterbi decoding has certain desirable properties:

- 1) The number of computations is proportional to 2^K , independent of rate.
- 2) Decoding can proceed at the data rate if the comparisons are done in parallel; therefore, the decoder does not have to be 10 to 20 times faster than the data rate, as in sequential decoding.
- 3) No block-tail structure is necessary; data may be continuous.

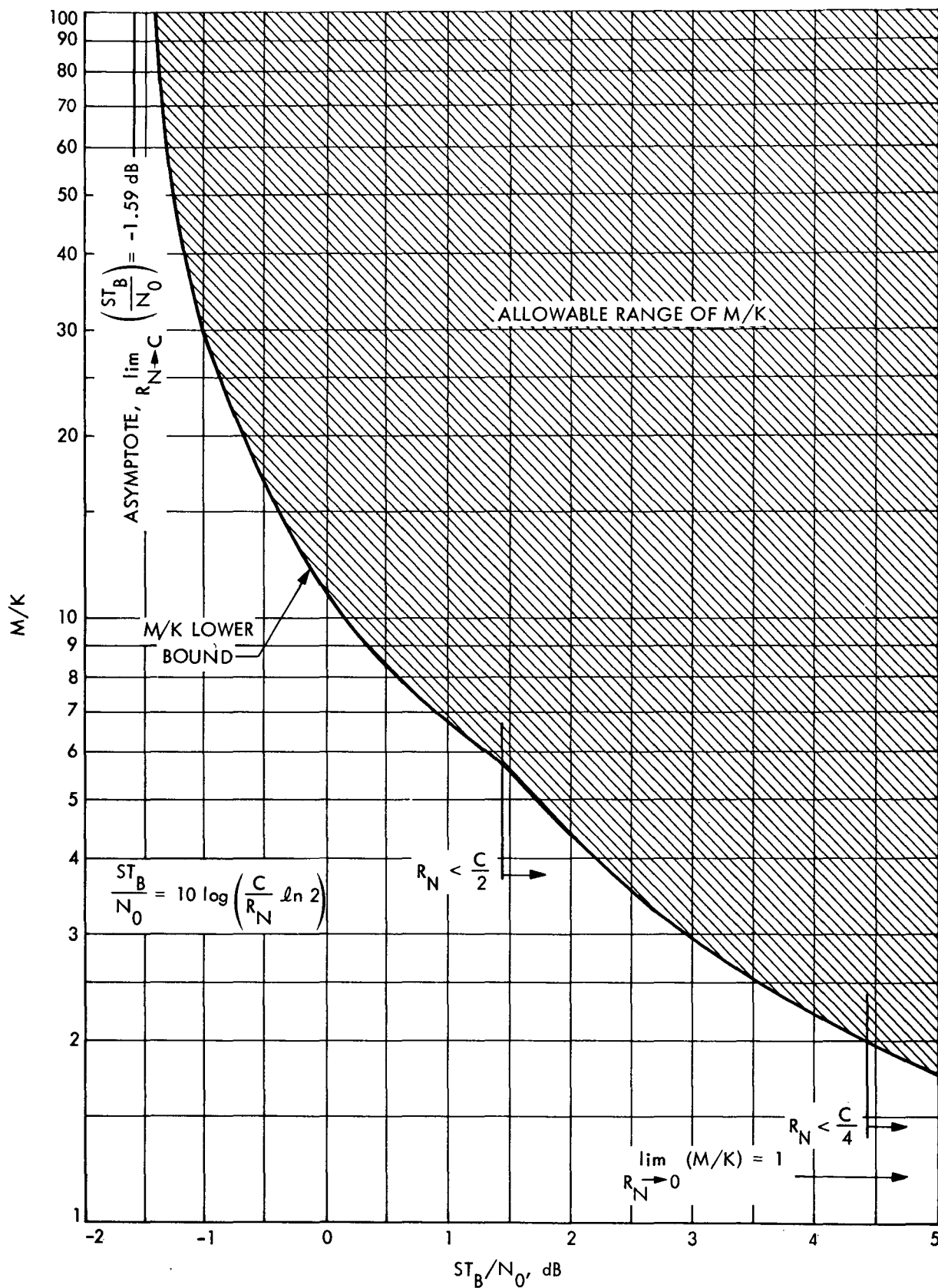


Fig. 5-17. Range of the number of constraint lengths for truncation in a Viterbi decoder which gives an additional probability of error less than P_{BE} .

- 4) Time to decode is fixed; therefore buffering is unnecessary.
- 5) Performance is typically 1 dB better than for biorthogonal codes of the same complexity.

Since decoder size grows as 2^K , Viterbi decoding is limited to short constraint length codes; hence, it becomes vital to select a "good" code. With the long constraint lengths typical of sequential decoding, less care can be taken in code selection, and better error rates can be obtained. Ironically, Viterbi decoding is most useful at high data rates. Sequential decoding, although less than optimal, is more useful at low data rates where machine speed can be 10 to 20 times faster than the bit rate, and advantages can be gained by the choice of large K .

Viterbi (reference 5-18) has derived very tight bounds on the probability of error for Viterbi decoding. These bounds must be evaluated numerically. This has been done in reference 5-19 and the results are shown in Figures 5-18 and 5-19. The difference between simulations and the error bounds is primarily due to a finite quantization loss of about 0.2 dB.

Heller and Jacobs (reference 5-19) have shown that, for a fixed constraint length, channel efficiency is increased by roughly the increase in channel capacity when the rate is decreased. Thus, performance improvement may be gained by increasing the bandwidth expansion if the effects on carrier tracking performance are small. The improvement, however, is not in proportion to the bandwidth expansion ratio. Thus, for a change from rate $1/2$ to rate $1/3$, the gain is between 0.3 to 0.5 dB, considerably less than $10 \log 3/2 = 1.76$ dB. If K is allowed to vary as well, Heller and Jacobs' results indicate that performance is approximately equal for encoders of equal complexity. Thus, the number of distinguishable states of an encoder is $2^{b(K-1)}$, so that the encoders $K = 3$, $R_N = 2/3$, and $K = 5$, $R_N = 1/2$ each have 16 states. The performance of these encoders differs by only 0.2 dB for equally good codes. Because of the steepness of the probability of error versus E_b/N_0 curves for a coded system ($E_b = ST_B$), small losses due to phase jitter in the carrier tracking loop cause large adverse changes in the received bit error rate. For a data rate $1/T_B \gg 2B_{LO}$, errors generated during Viterbi decoding will occur over a small enough time that the phase error in the carrier loop, ϕ ,

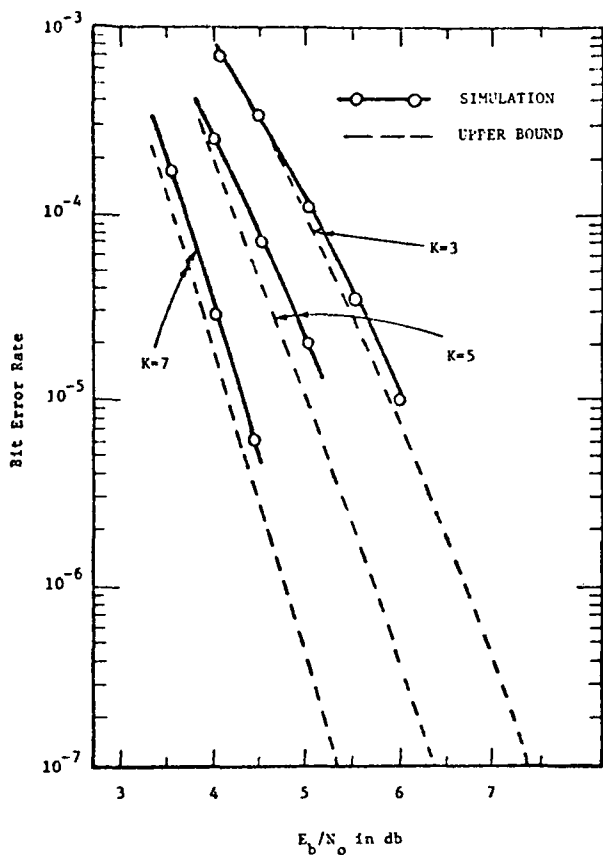


Fig. 5-18. Bit error rate versus E_b/N_0 for rate 1/2 Viterbi decoding. 8-level quantized simulations with 32-bit paths, and infinitely finely quantized transfer function bound, $K = 3, 5, 7$. (Source: Reference 5-19, with permission)

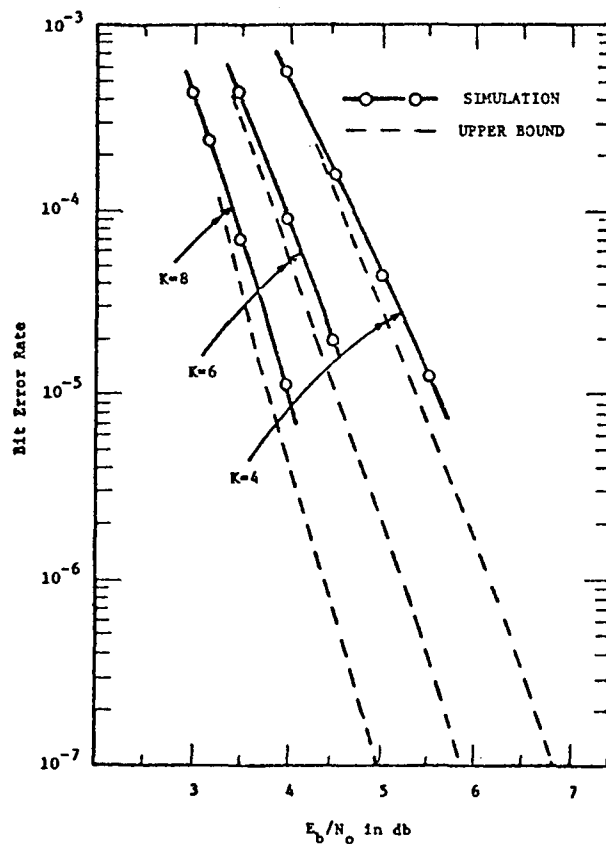


Fig. 5-19. Bit error rate versus E_b/N_0 for rate 1/2 Viterbi decoding. 8-level quantized simulations with 32-bit paths, and infinitely finely quantized transfer function bound, $K = 4, 6, 8$. (Source: Reference 5-19, with permission)

will remain constant and errors will be independent. Following reference 5-19, if the probability of bit error is written as

$$P_B = f(E_b/N_0) \quad (5.3-26)$$

then

$$P_B(\phi) = f(E_b \cos^2 \phi / N_0) \quad (5.3-27)$$

For a distribution $p(\phi)$, the system bit error probability is

$$P_B' = \int_{-\pi}^{\pi} p(\phi) P_B(\phi) d\phi \quad (5.3-28)$$

Figure 5-20, taken from reference 5-19, plots P_B' versus E_b/N_O for a $K = 7$, $R_N = 1/2$ code and a second-order tracking loop,

where

$$p(\phi) \approx \frac{\exp(\eta \cos \phi)}{2\pi I_0(\eta)} \quad \eta \gg 1$$

where η = loop SNR. We see that P_B' is a very strong function of η . The difference at a constant error rate between the curves for finite η and $\eta = \infty$ is the one-way radio loss for Viterbi decoding of the convolutional code. Radio loss, η_{RL} , will be discussed in paragraph 5.4.5.

5.3.3.2.1.3 Comparison of Sequential and Viterbi Decoding. In Table 5-4, we consider some of the tradeoff considerations in selecting sequential or Viterbi decoding methods (see reference 5-19).

Heller and Jacobs have further shown that since sequential decoding is very sensitive to receiver AGC variations, it may only be practical to have hard decision capability ($Q = 2$). Under these circumstances, at $P_B \sim 10^{-5}$, a $K = 5$ to 7 , $R_N = 1/2$, $Q = 8$ Viterbi decoder has performance equivalent to a $K = 41$, $R_N = 1/2$, $Q = 2$ sequential decoder, although for $P_B < 10^{-8}$, the sequential machine is superior.

5.3.3.3 Concatenated Coding. The acceptable ST_B/N_O threshold for a telemetry channel is controlled by the transmitted data with the lowest allowable error rate. If it happens that the required error rate for this data is significantly lower than the requirements for the bulk of the telemetry, transmission of the entire data stream at the lowest error rate may prove very costly in terms of power or data rate. Examples for which this situation may obtain include:

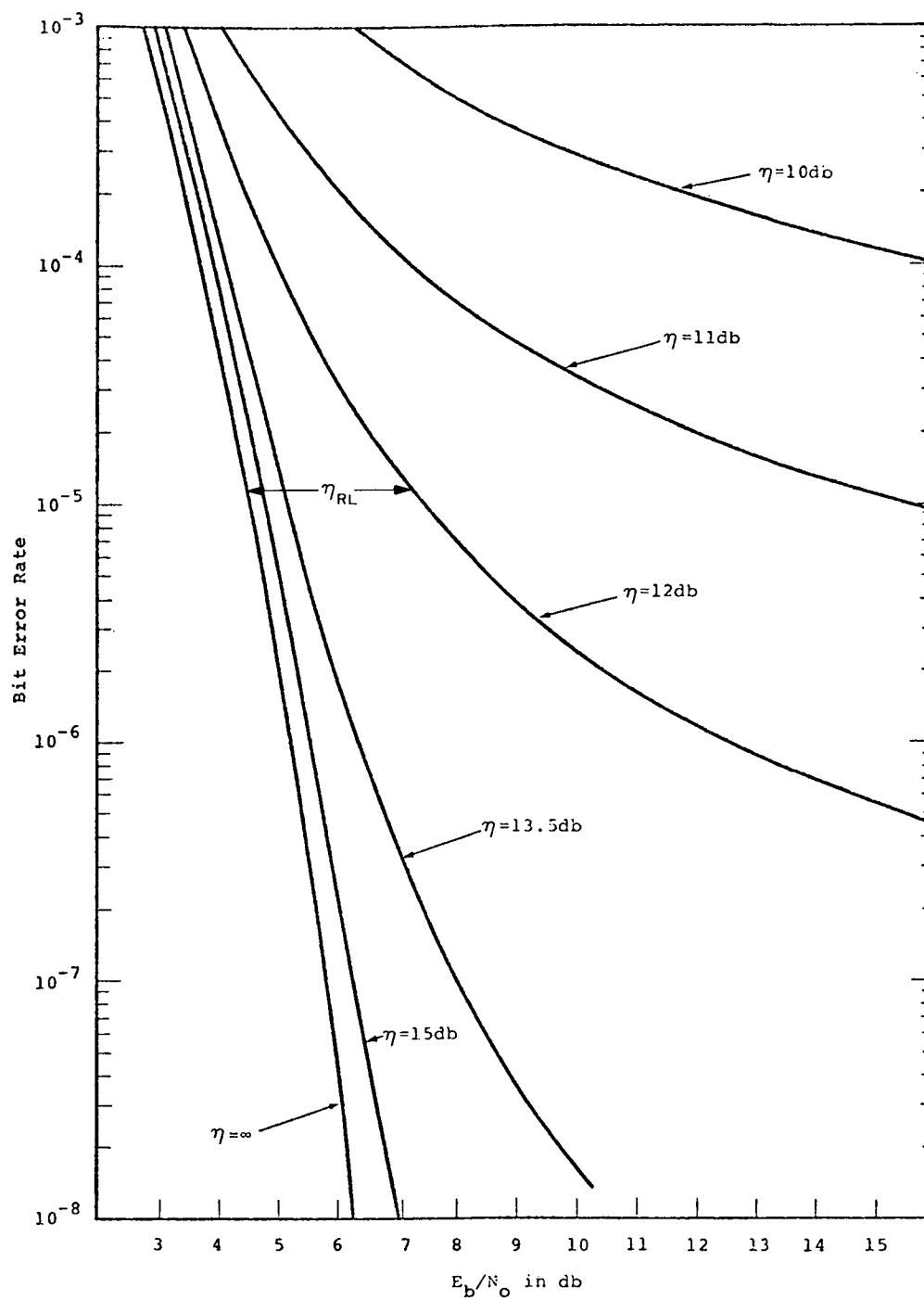


Fig. 5-20. Performance curves for rate 1/2; K = 7 Viterbi decoder with 8-level quantization as a function of carrier phase tracking loop signal-to-noise ratio η
(Source: Reference 5-19, with permission)

Table 5-4. Tradeoff considerations for Viterbi and sequential decoding methods

System Variables	Viterbi	Sequential
Bit Errors in Output	Short bursts of 10 to 20 bit errors	Bit error probability can be made extremely low, but errors come in bursts of up to L bits at a time
Decoder Delay	Short, less than $100 T_B$ delay	Output may be delayed many thousands of bit times
Transmitted Data Format	Continuous	In blocks with tail of no data added
Choice of Rate and Quantization	Decoder completely insensitive to these variables	Storage requirements strongly dependent on Q and R_N
Sensitivity to AGC of Receiver	Insensitive	Very sensitive for large Q
Sensitivity to Carrier Loop Phase Error	Shown in Figure 5-20	Much more sensitive due to steeper curve; also, phase varies over long error bursts, implying errors are not independent.

- 1) TV transmission may require error rates of 1 in 10^3 , while other experiments occupying a fraction of the channel may require 1 in 10^5 .
- 2) Only long and completely error-free data frames may be useful for certain experiments.
- 3) Certain key data may require extremely high confidence.

To alleviate this problem, at least three methods are available:

- a) Adding another telemetry channel for the low error rate data;
- b) Interplex modulation; and
- c) Concatenated coding.

The method of concatenated coding is illustrated in Figure 5-21. Concatenated coding allows the use of additional coding for part of the data without loading the remaining data by that additional redundancy. In its use of a combination of two separate coding schemes, concatenated coding requires little new theory, but much analysis must be performed to determine the proper inner and outer codes and decoding methods. The following are some of the considerations important in the use of concatenated coding:

- 1) Additional coding means additional redundancy or check bits at the cost of less transmitted information. Therefore, only few data which really need higher confidence should be protected by additional coding.

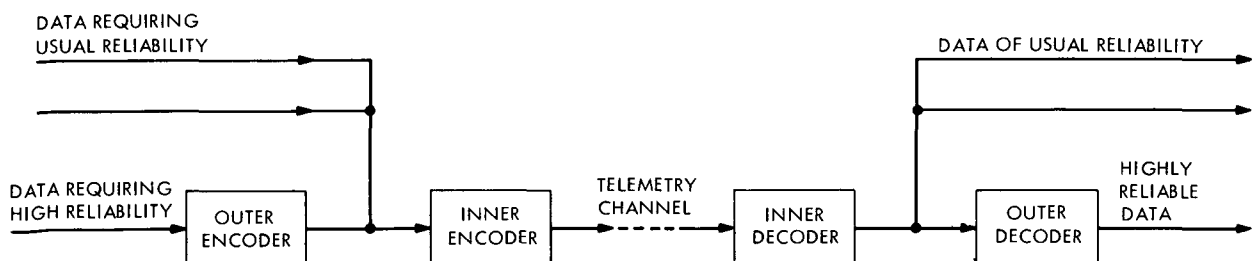


Fig. 5-21. Concatenated coding scheme

- 2) Increased data reliability can be gained as a trade-off between additional redundancy, encoding/decoding complexity and data deletion. If, for example, the deletion of a small amount of unreliable data can be permitted, the required redundancy and/or complexity decreases remarkably.
- 3) Simple algebraic coding/decoding can be used effectively for outer coding.
- 4) The outer code has to be of a burst-error correcting type because the errors of the inner decoder are bursts of wrong codewords or codeframes of the inner code.
- 5) For effective outer coding, it is of advantage if the codewords of the inner code have fixed locations in the data format, i. e., if inner coding and data frames are synchronous. The parameters of the outer code depend strongly on the inner coding scheme. For example, for convolutional/sequential decoding in the inner decoding scheme, the error probability of decoded frames, in principle, can be made as small as desired by making the constraint length of the convolutional code long enough. The problem is that some frames might not be decoded due to time limitations of the sequential decoder. For the outer decoder, this has the advantage that the location of the erroneous elements are known. In this case, the erroneous elements are called erasures rather than errors. Erasures can be corrected with much less redundancy and/or decoding complexity than errors.

If maximum likelihood decoding (block decoding or convolutional/Viterbi decoding) is chosen for inner decoding, all data are decoded and errors are undetected. In this case, a more complicated error correcting procedure and/or error detection (with deletion of erroneous data) has to be used in the outer decoding.

The application of concatenated coding is discussed in references 5-20 through 5-27.

A concatenated coding scheme has been used for the Mariner Mars 1971 Infrared Interferometer Spectrometer (IRIS) experiment, where an improvement of two orders of magnitude in probability of bit error was achieved. Reference 5-27 discusses the coding scheme used.

A basic block diagram of the outer decoder with detector-corrector processing is given in Figure 5-22, taken from the reference. Typical input/

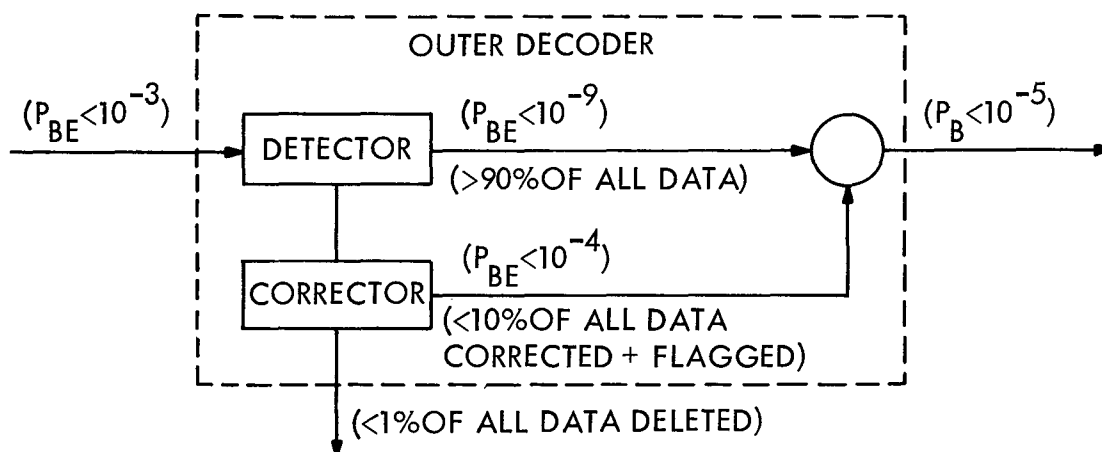


Fig. 5-22. Outer decoder with detector-corrector processing (MM'71 IRIS) (P_{BE} = bit error probability)

output values are included in parentheses. Figure 5-23 shows the bit error probability vs signal-to-noise ratio for the ($n = 6$, $M = 2$, $k = 6$) generalized Hamming outer code and a biorthogonal ($n = 32$, $k = 6$) inner code used, where n is the number of symbols formed from k information bits and M is the number of check elements in the outer code.

5.3.4 Data Synchronization

The accuracy of data symbol synchronization is vital to the quality of telemetry data finally presented to the users. There are two aspects to synchronization accuracy. The first involves the method by which data is formatted and synchronized on the spacecraft before transmission to Earth. The second aspect is the quality of data synchronization obtained on the ground and how the data is deformatted.

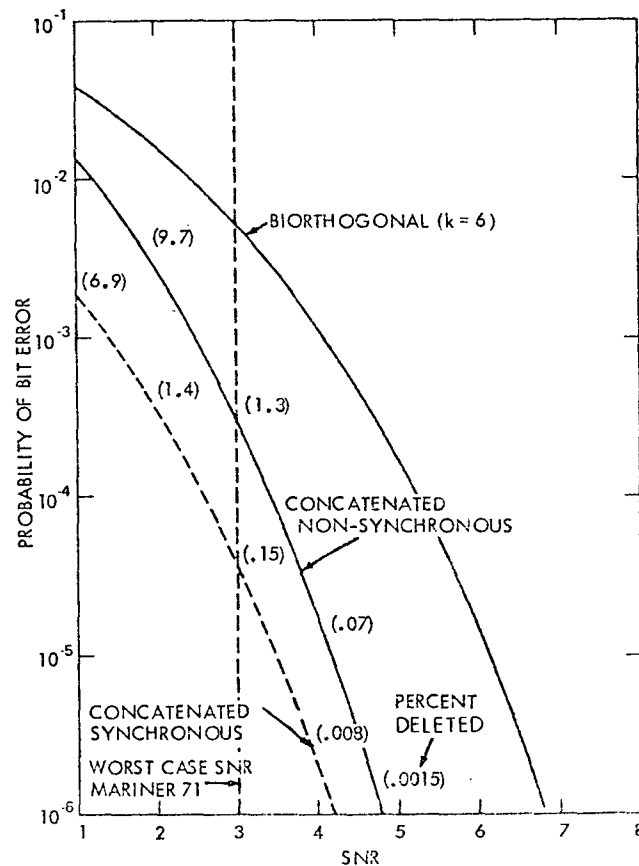


Fig. 5-23. Probability of bit error after outer decoding vs signal-to-noise ratio, SNR, for concatenated synchronous and nonsynchronous systems as referenced to a biorthogonal ($k = 6$) code. (MM'71 IRIS)

5.3.4.1 Types of Synchronization. In the spacecraft, data is obtained from sensors, digitized, formatted into frames and outputted in a serial bit stream for transmission to Earth. To deformat the data properly, some form of frame sync is needed. This is usually a PN code word which is added to the beginning of the frame. In some cases, subformatting is provided, and PN sync codes may occur more than one time per frame.

Once the data is formatted and processed into a serial data stream, the data bits are modulated onto subcarriers, or they are mapped into code symbols and then modulated onto subcarriers. If data is to be sent uncoded, no special synchronization code is required, but it is subject to one condition;

there must be a minimum rate of transitions between "1's" and "0's" in the data to assure acquisition of symbol sync and good efficiency in the SAA at the DSS. A minimum transition density of 10 to 20 percent is recommended. If long sequences of all "1's" or all "0's" are anticipated, such as an all black or all white TV picture or a computer memory readout where some of the memory is blank, some sort of transition generator should be added to the flight data system to aid in the symbol synchronization at the DSS. However, care must be taken to prevent a loss in transition density in more typical data.

If data is block coded, the flight data system must provide, in addition to the data bits, a bit sync clock pulse, a word sync clock pulse, and code symbol sync clock pulse.

If the block code chosen uses data words k bits long, the word sync rate R_{WS} must be $1/k$ of the bit sync rate, R_B .

$$R_{WS} = \frac{1}{k} R_B \quad (5.3-29)$$

A symbol sync pulse must be provided which is exactly $M = 2^k - 1$ times the word sync rate.

$$R_{SS} = \left(2^k - 1 \right) R_{WS} = \left(\frac{2^k - 1}{k} \right) R_B \quad (5.3-30)$$

If a convolutional encoder of rate $1/v$ and constraint length K is used, the spacecraft data system must supply a symbol sync pulse to the coder which is exactly v times the bit rate. Then,

$$R_{SS} = v R_B \quad (5.3-31)$$

5.3.4.2 Use of Synchronization in Ground Processing. When the transmitted symbols are received on Earth, the various synchronizations that were put into the data must be recovered and used.

After the carrier and subcarrier have been demodulated, the received data symbols plus noise are outputted from the SDA. The SSA accepts this output, integrates the data-plus-noise, and attempts to estimate the original synchronization from the integrated data. The noise in the SDA output tends to mask the data transitions so that an imperfect symbol sync reference will be derived. Thus, the efficiency of symbol synchronization is a function of the signal energy per bit, ST_B/N_O . Furthermore, during the times when there are no transitions in the symbol stream, the reference oscillator may drift out of sync with the symbols. Thus, the efficiency of symbol sync is also a function of the data transition density.

The symbol synchronization reference is used in the correlation detection of data bits or code words. If symbol sync is imperfect, the error probabilities will be worse than that predicted in equations (5.3-5) and (5.3-13).

If data is uncoded, the SSA will produce estimates of the data bits. These bits are formatted by the TCP for transmission from the stations to the Project. If the data is block coded, the data symbols are passed on to the BDA. The BDA must obtain word synchronization by looking for high correlation with words in the block code dictionary. Again, imprecise knowledge of word sync will degrade the error probability of equation (5.2-13).

If data is convolutionally encoded, the code symbols from the SSA are presented to the DDA for detection. If the symbols are not perfectly synchronized, the error rate of the convolutional decoder is increased. The DDA must generate branch synchronization and block synchronization from the incoming symbols. Block synchronization is generated from the tail appended to the data. Branch synchronization is generated automatically if the code is properly chosen. Sometimes advantage may be gained by the addition of a sync sequence to the encoder output on the spacecraft. Reference 5-31 discusses the conditions required for branch synchronization.

For a Viterbi decoder, no block sync is necessary. Branch sync considerations are discussed in reference 5-31.

After the data bits have been detected or decoded they are formatted and prepared for transmission to the Project over one of the communications lines used by the DSN. These lines are the Ground Communications Facility

(GCF), the High Speed Data Link (HSDL), the Wide-Band Data Link (WDL), and the Microwave Link. In some cases, the data is put into a buffer so that the Project's computers may accept the data bits at the proper time in the computation cycle. The computer must first obtain frame sync from the data bits, and then deformat the data. To do this, the computer algebraically compares the received data sequence with the known frame sync code. Since the data bits will have some error probability associated with them, the frame sync algorithm in the computer should declare the data in sync even though there are a few disagreements between the received bits and the sync code word.

Though the data may be in sync, it is necessary for the computer to look at each sync code word to ensure that the data remains in sync. Momentary interruptions in the data occur. For example, the data during spacecraft tape recorder playback will be momentarily interrupted when tape drive direction is changed. A threshold number of acceptable discrepancies in the sync word to maintain frame sync should be based on a knowledge of the expected bit error probability of the incoming data bits, and the probability that frame sync may be lost due to data interruptions.

The probability of frame sync loss strongly affects the quality of data returned to the user.

5.3.5 Relating Channel Error Rates to User Error Rates

To predict the quality of data ultimately handed the user, it is necessary to construct a model of error sources, such as shown in Figure 5-24. Physical sensors of analog data can be thought of as having some information bandwidth, and as introducing some kind of noise into its output.

In digitizing, the analog data is quantized into one of 2^k discrete levels. This quantization represents a bandwidth upper limit on the information obtained from the sensors. Each sensor is sampled at some rate which depends on the number of sensors to be sampled and the channel transmission rate. This sampling also limits the information bandwidth.

Data handling errors on the spacecraft occur primarily in the storage of digital or analog data on a tape recorder. Good design practice indicates that if recording errors are to contribute a negligible amount to the

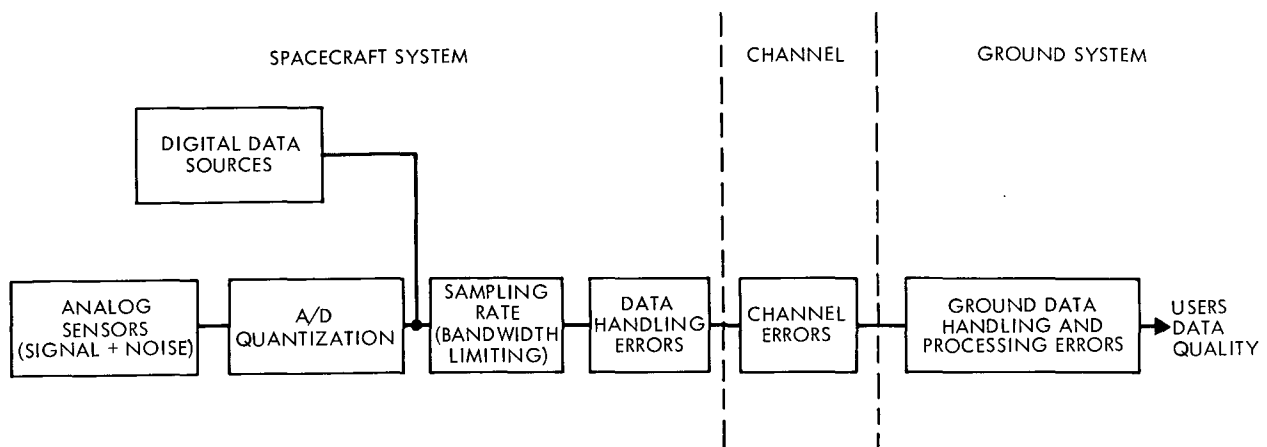


Fig. 5-24. Data quality model

user's error rate, they should be at least an order of magnitude less than the channel's error rate.

Channel errors are introduced in the process of detecting the transmitted signal imbedded in noise. The error probabilities which have been discussed parametrically in the preceding sections depend on the received signal-to-noise ratio, and the ability of the ground system to recover data synchronization.

Ground data handling errors are of the following two types:

- 1) Bit errors are introduced in the transmission of data between the tracking station and Network Control Center
- 2) Synchronization errors and loss of data occur due to computer limitations.

To be negligible, bit errors introduced in the ground communications process should be at least an order of magnitude lower than the channel error rate.

Loss of data due to buffer overflows should be minimized by matching the data rates and data processing required to the capabilities of the computer. Ideally, the ability to deliver a prescribed quantity of data should be a specification on the ground system performance.

Loss of data due to synchronization errors can occur when bit errors in the data going into the computer cause the computer to:

- 1) Think it has frame sync when in fact it does not;
- 2) Think it has lost frame sync when in fact the data is still synchronized;
- 3) Or lose frame sync entirely.

These errors are a function of the type of channel coding used and the bit or word error rates associated with the coding.

5.3.5.1 User Error Rates as a Function of Channel Error Rates. One important aspect of the problem of modeling the user's data quality is to relate the bit or word error probability of the channel to the bit or word error probability of the user. What follows, assumes that all errors other than channel errors are negligible.

The user may wish to specify his data quality in terms of an average bit error rate. If the channel is uncoded, then the channel bit error rate is identical to the user bit error rate, and the bit errors will be independent of each other.

$$P_{B \text{ USER}} = P_{B \text{ CHANNEL}} \quad (5.3-32)$$

The user may wish to specify his data quality in terms of an average word error rate for data of some length, M. If the channel is uncoded, then the probability that the word will be in error is one minus the product of the probability that each bit is correct.

$$P_{W \text{ USER}}(M) = 1 - (1 - P_{B \text{ CH}})^M \approx MP_{B \text{ CH}} \quad (5.3-33)$$

assuming

$$P_{B \text{ CH}} \ll 1$$

For the uncoded channel, these word errors will be independent of each other.

If the channel is block encoded, the detection of 5- or 6-bit words is performed. Thus, an entire block of 5 or 6 bits is correct, or the word is in error and one or more bits are in error. The coded word errors are independent of each other, but the bit errors come in bursts. A user may be interested in data words whose length is different than that of the block coded word transmitted as a block of code symbols. Consequently, the user's data word error rate, ϵ_D , will be different from the block coded word error rate, ϵ_C . The ratio of ϵ_D/ϵ_C depends on the length of the data word, the block coded word, the probability of coded word errors, and the relative synchronization of data words with code words. In general, the lowest user data word error rate occurs when the number of code words which overlap a data word is minimized.

If the data channel is block coded and the user is interested in the average bit error rate, the bit error rate is given by equation (5.3-14).

5.4 TELEMETRY SYSTEM EFFICIENCY

Telemetry channels for deep space vehicles have typically been power limited rather than bandwidth limited. For this reason, the efficiency of a telemetry system design is measured by how much power is required to achieve a given data rate with a given maximum bit error rate. The design and performance of a telemetry system can be conveniently referenced to the ratio of total received power to system noise spectral density, P_T/N_O . This parameter was discussed in detail in section 2. There it was shown that P_T/N_O is completely determined by the spacecraft transmitter power, the spacecraft and ground station antenna gain patterns, ground system noise temperature, and the orientation of the spacecraft as a function of its mission trajectory.

In this section, it will be shown how to relate the performance of a particular telemetry system to the independent variable P_T/N_O , and how a telemetry system may be optimized as a function of P_T/N_O .