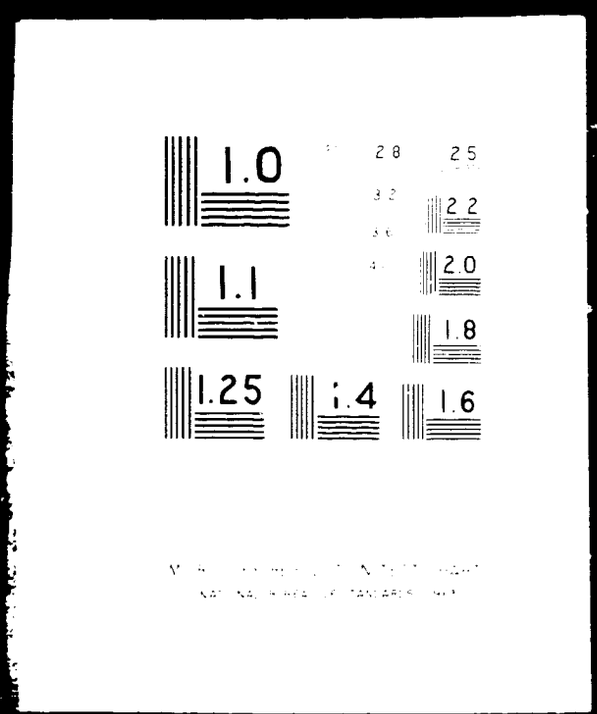
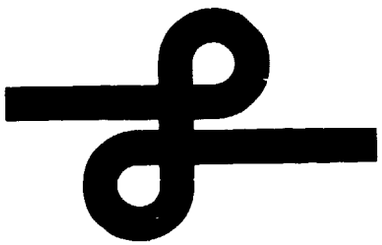
1 OF 2

N73 20889 UNCLAS



STUDY OF
INFORMATION TRANSFER OPTIMIZATION
FOR COMMUNICATION SATELLITES

FINAL REPORT



LINKABIT CORPORATION

10453 ROSELLE STREET
UNIVERSITY INDUSTRIAL PARK
SAN DIEGO, CALIFORNIA 92121

UNCLASSIFIED

NASA CR No. 114561
Available to the Public

STUDY OF INFORMATICN TRANSFER OPTIMIZATION FOR COMMUNICATION SATELLITES

FINAL REPORT

By

Joseph P. Odenwalder
Andrew J. Viterbi
Irwin Mark Jacobs
Jerrold A. Heller

January 3, 1973

Prepared under Contract No. NAS2-6810

by

LINKABIT Corporation
San Diego, CA 92121

for

AMES RESEARCH CENTER
NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

UNCLASSIFIED

TABLE OF CONTENTS

TABLE OF FIGURES

TABLE OF TABLES

1.0 Introduction

This report presents the results of a study of source coding, modulation/channel coding, and systems techniques for application to teleconferencing over high data rate digital communication satellite links. Simultaneous transmission of video, voice, data, and/or graphics is possible in various teleconferencing modes and one-way, two-way, and broadcast modes must be accommodated.

Power, bandwidth, complexity and reliability (including insensitivity to channel degradations) are key cost factors in any communication system and become especially important for those teleconferencing modes which rely on small, possibly transportable ground terminals. Acceptable fidelity to the user is necessary for each of the types of source material. Since increased fidelity generally implies increased cost, fidelity and the cost parameters are presented parametrically whenever possible.

The satellite channel is considered first in Section 2.0. A model including filters, limiter, a TWT, detectors, and an optimized equalizer is considered in detail. A complete analysis is presented for one set of system assumptions which exclude nonlinear gain and phase distortion in the TWT. This analysis is suggestive of the advantages achievable by signal design and equalization

and the complexity required, but must be specialized to actual satellite transponder models. The present computer programs permit considerable specialization but must be augmented to provide for more elaborate models.

Modulation, demodulation, and channel coding are considered in Section 3.0 based on an additive white Gaussian noise channel model which is an idealization of the equalized channel of Section 2.0. Amplitude and phase modulated systems are considered together with Viterbi-decoded convolutional codes of several rates and constraint lengths. The low error probability requirements of data transmission are approached via concatination with a simple, heavily interleaved, convolutional outer code.

Source coding with emphasis on video data compression is considered in Section 4.0. The experimental facility utilized by LINKABIT to test promising techniques is fully described. With this system, subjective evaluation is aided by the introduction of a unique split-screen presentation technique in which the compressed and uncompressed versions are presented side-by-side. Promising techniques based on 2-and 3-Dimensional Hadamard transformers are described and evaluated. Voice and graphics compression are treated more briefly since th[illegible] have much smaller impact on data rates.

Section 5.0 presents system considerations involved in effectively utilizing the source and channel coding techniques of earlier sections. These considerations, of course, are heavily influenced by the desired teleconferencing modes. Emphasis is placed on TDMA because of its flexibility, efficiency, and compatibility with the digital transmission techniques and the reduction of degradation due to transponder nonlinearities.

Conclusions and recommendations for future activity are presented in Section 6.0.

2.0 THE SATELLITE CHANNEL

Figure 2.0.1 gives a block diagram of the satellite transponder. Three characteristics of this channel impose particular properties and constraints on a communication system. In order of importance these are

1) The satellite transponder nonlinearities which may preclude the use of signal shaping or pre-emphasis techniques for pre-compensating for the linear channel distortion. If the TWT is operating in an approximately linear range, then since the uplink SNR is usually much higher than the downlink SNR and the first transponder IF filter is generally much wider than the second IF filter, we may limit consideration to signals (noiseless pulse trains) beginning at the input of the second IF filter. If the TWT is not operating in a linear range and a nonlinear gain is introduced, this gain can be modeled as a memoryless nonlinearity following the second IF filter, and its effects can be incorporated into the correlation receiver. If the phase distortion is also significant, the problem is much more difficult, and simulation would be more useful than analysis. Here we use the linear approximation.

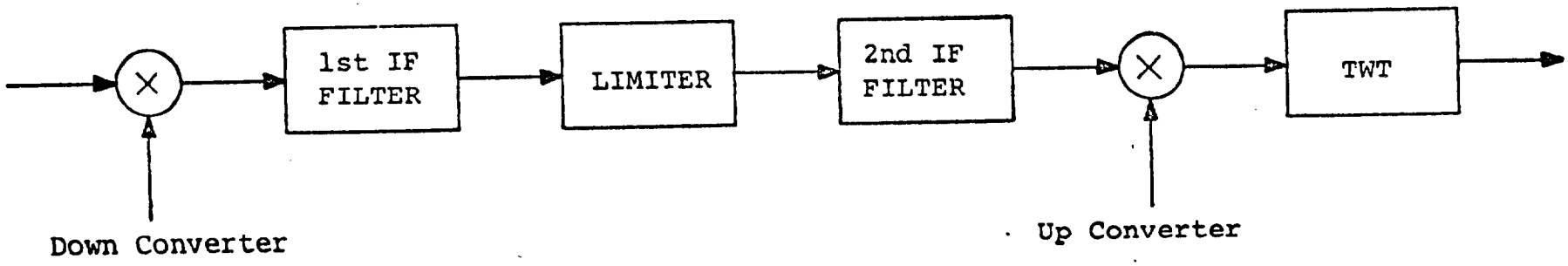


Figure 2.0.1. Satellite Transponder

2) The intersymbol interference due to narrowband filtering which will be nearly time-invariant in contrast to the situation in digital communication over switched telephone lines. Hence, adaptive equalization will not be required, and attention should be devoted to optimizing a fixed equalizer.

3) The premium placed on downlink power. The appropriate energy parameter is the energy at the output of the second IF filter; that is, after attenuation by the filter.

We consider here primarily binary PSK. However, the results hold also for quaternary PSK provided the (2nd IF) filter transfer function is symmetrical about its center frequency. This guarantees that there will be no intersymbol interference between the two quadrature channels and consequently that each of the two channels can be treated as a separate and independent BPSK channel. In all cases we assume that $G(\omega)$, the filter transfer function, is rational in ω.

We treat two cases which represent the two extremes of very simple and very complex modem analog front ends. These are

1) an integrate-and-dump (I&D) filter,

(2) a whitened matched filter (WMF) which is the optimum demodulator.

In each case we determine the optimum equalizer to minimize the error probability for uncoded transmission.

2.1 Discrete Transfer Functions of Intersymbol Interference

2.1.1 I&D Filter

For an input pulse of duration T, the transmitted signal transfer function is

$$\frac{1-z^{-1}}{i\omega} \; G(\omega) \qquad , \; z = e^{i\omega T}$$

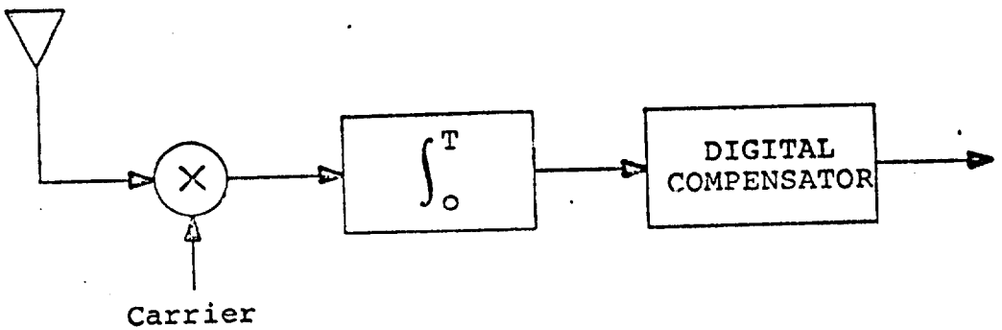
where $G(\omega)$ is the transponder filter transfer function. The I&D demodulator filter has transfer function

$$\frac{1}{T} \qquad \frac{1-z^{-1}}{i\omega}$$

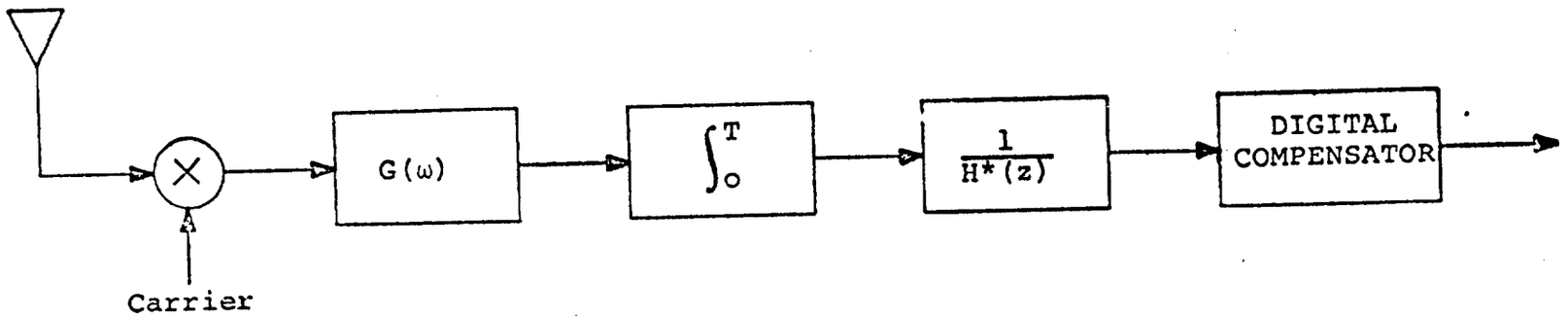
and is therefore the matched filter for the square pulse without filtering by $G(\omega)$. The discrete transfer function for the overall filtering operation as shown in Figure 2.1.2 (a) is

$$H^*(z) = \frac{(1-z^{-1})^2}{T} \left[\frac{G(\omega)}{(i\omega)^2}\right]^* \qquad (2.1.1)$$

where the superscript * represents the sampling operation.

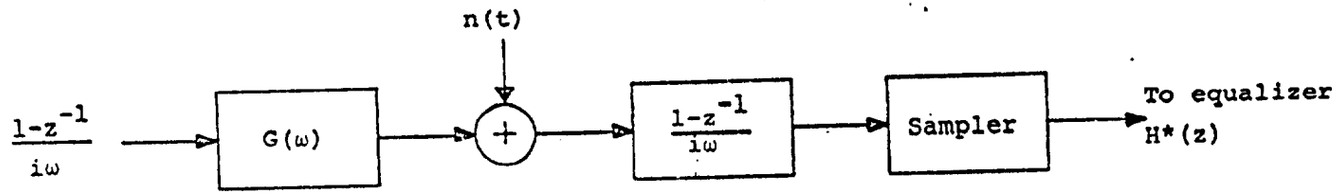


(a) I&D Filter

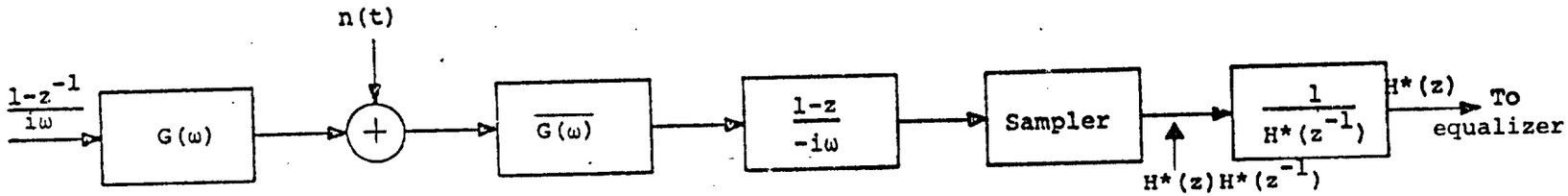


(b) Whitened Matched Filter

Figure 2.1.1. Alternate MODEM Front Ends



(a) I&D Filter



(b) Whitened Matched Filter

Figure 2.1.2. Filter Discrete Transfer Functions

The overall discrete transfer function can be computed in the following manner:

a) Find roots of $G(\omega)$ and perform a partial fractions expansion

$$\sum_j \frac{C_j}{i\omega+\alpha_j} \quad \text{where } C_j \text{ and } \alpha_j \text{ are complex.}$$

b) Convert each root to discrete form by the operation (we assume all roots are simple)

$$\frac{(1-z^{-1})^2}{T}\left[\frac{C_j}{(i\omega+\alpha_j)(i\omega)^2}\right]^* = \frac{C_j\, z^{-1}}{\alpha_j}\left[\left(1-\frac{1-e^{-x_j}}{x_j}\right)+\left(\frac{1-e^{-x_j}}{x_j}\right)^2\frac{z^{-1}}{1-e^{-x_j}z^{-1}}\right]$$

$$\text{and } x_j = T\alpha_j \qquad (2.1.2)$$

c) Combine terms to obtain $H^*(z)$. Note that the order of $H^*(z)$ is the same as that of $G(\omega)$.

A subroutine has been written to perform this operation for any rational filter $G(\omega)$ up to degree 20. This is entitled and defined as follows:

DlSTF (CCOF, DCOF, M, DWT, HNUM, IHNUM, HDEN, IHDEN)

Input parameters:

CCOF and DCOF are polynomials representing numerator and denominator of G(s), where $s = i\omega$, normalized such that

the 3 dB bandwidth W = 1/2T.

M = degree of denominator

DWT = actual 2WT

HNUM and HDEN = numerator and denominator of $H^*(z)$

IHNUM-1 and IHDEN-1 = degrees of these polynomials

We note also that the I&D filter in response to additive white Gaussian noise of one sided density N_o produces independent noise samples of variance $N_o/2T$.

2.1.2 Whitened Matched Filter

The optimum demodulator is a matched filter whose transfer function, matched to the distorted pulse, is

$$\frac{1-z}{-i\omega} \quad \overline{G(\omega)}$$

where the overbar indicates the complex conjugate.

Thus, the overall discrete transfer function of the modulator plus the demodulator sampled matched filter is

$$\left[\frac{\left(1-z^{-1}\right)\left(1-z\right)}{-\left(i\omega\right)^2}\left|G(\omega)\right|^2\right]^* = H^*(z)\,H^*\left(z^{-1}\right)$$

or

$$H^*(z)\,H^*\left(z^{-1}\right) = \left(1-z^{-1}\right)^2 z\left[\frac{\left|G(\omega)\right|^2}{(i\omega)^2}\right]^* \qquad (2.1.3)$$

Noting the similarity between (2.1.3) and (2.1.1) we may use the following procedure to compute $H^*(z)\ H^*(z^{-1})$:

a) Find the roots of $G(\omega)$ and make a partial fraction expansion of $|G(\omega)|^2$ in the roots of $G(\omega)$ (LHP poles)

b) Convert each root into discrete form by (2.1.2)

c) Combine terms to obtain $K^*(z)$

d) Steps (a) - (c) on the roots of $\overline{G(\omega)}$ (RHP poles) would produce $K^*(z^{-1})$. Therefore,

$$H^*(z)H^*(z^{-1}) = K^*(z) + K^*(z^{-1}) = H_T^*(z) \qquad (2.1.4)$$

We note that $H^*(z)$ is the modulation discrete transfer function and $H^*(z^{-1})$ is that of the matched filter. Thus the noise sequence is effectively filtered by $H^*(z^{-1})$ and hence is no longer white. We may whiten it by passing it through $1/H^*(z^{-1})$ or $1/H^*(z)$ (see Fig. 2.1.3.).However, the first is unrealizable, except with infinitely long delay, while the second is realizable. The only difference in the two cases is that the intersymbol interference in the first case is caused by preceding message symbols, while in the second it is caused by succeeding message symbols. The performance is the same for both.

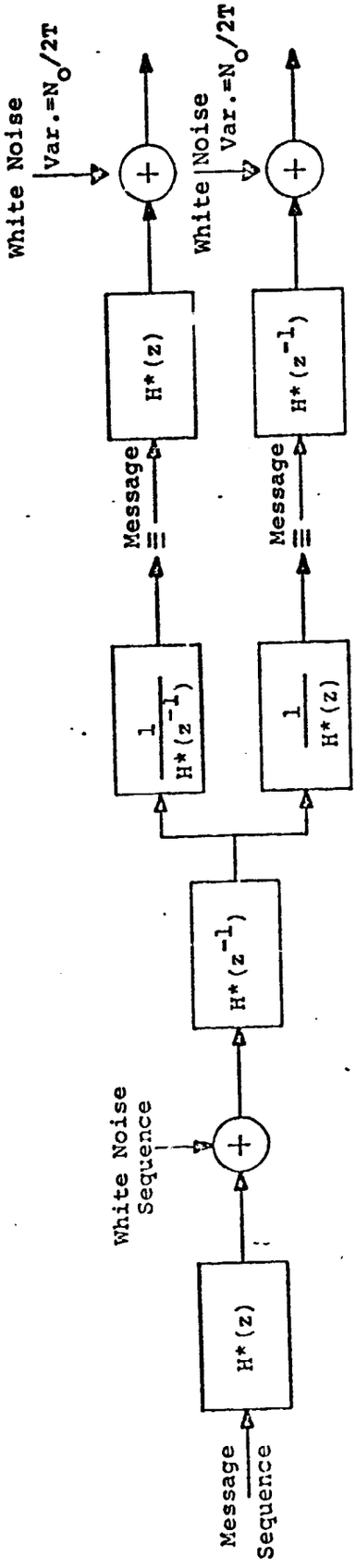


Figure 2.1.3. Alternate Implementations of Whitened Matched Filter

Finally, to determine $H^*(z)$ or $H^*(z^{-1})$ we must factor (2.1.4). The denominator factors are known, but not the numerator. This is most easily found by multiplying together all the numerator roots with the unit circle. We note also that the order of $H^*(z)$ is the same as that of $G(\omega)$.

A subroutine for calculating the whitened matched filter $H^*(z)$ for any rational filter $G(\omega)$ up to degree 20 has been written. It is entitled
DISMF (CCOF, DCOF, M, DWT, HNUM, IHNUM, HDEN, IHDEN)
where the terms are defined as for the I&D filter (Sec. 2.1.1).

The energy transmitted by the satellite is that of a binary PSK signal attenuated by $G(\omega)$. Thus, if E_b were the received bit energy for $G(\omega) = 1$, the true received bit energy is

$$E_b^{\,1} = E_b \int_{-\infty}^{\infty} \left| \frac{\tilde{G}(\omega)}{i\omega} \right|^2 (1-z^{-1})\ (1-z)\ d\omega/2\Pi$$

But this is just the energy at the matched filter output and consequently by Parseval's Theorem, this can also be expressed as

$$E_b^1 = E_b \sum_{k=0}^{\infty} Z_k^2$$

where Z_k is the k^{th} coefficient of $H^*(z)$.

2.2 Error Probability Calculation

The error probability for BPSK in intersymbol interference is

$$P_E = \frac{1}{2}\left[\Pr\left(-Z_1 + \sum_{i=2}^{K} \alpha_i Z_i + \frac{n}{\sqrt{E_b/T}} > 0\right) + \Pr\left(Z_1 + \sum_{i=2}^{K} \alpha_i Z_i + \frac{n}{\sqrt{E_b/T}} < 0\right)\right]$$

where Z_i is the i^{th} coefficient of $H^*(Z)$, n is the noise on the given symbol, and α_i is a unity magnitude variable with a positive polarity when the $(i-1)^{th}$ previous input symbol was one and a negative polarity when it was zero.

Thus

$$P_E = \frac{1}{2}\left[1 - \Pr\left(-Z_1 < \sum_{i=2}^{K} \alpha_i Z_i + \frac{n}{\sqrt{E_b/T}} < Z_1\right)\right]$$

Since the interfering symbols are all independent and equally likely to be 0 and 1, and independent of the noise, we have, letting

$$U = \sum_{i=2}^{K} \alpha_i Z_i + \frac{n}{\sqrt{E_b/T}}$$

$$P_E = \frac{1}{2}\left[1 - \int_{-Z_1}^{Z_1} p_U(u)\, du\right]$$

$$= \frac{1}{2}\left[1 - \int_{-\infty}^{\infty} \frac{\sin Z_1 u}{\pi u}\, \phi_U(u)\, du\right]$$

$$= \frac{1}{2}\left[1 - \int_{-\infty}^{\infty} \frac{\sin Z_1 u}{\pi u} \prod_{i=2}^{K} \cos Z_i u \left(e^{-\frac{\sigma^2 T u^2}{2E_b}}\right) du\right] \qquad (2.2.1)$$

where $\sigma^2 = N_o/2T$

and $\phi_U(u)$ is the characteristic function of u.

Thus, the first step in the calculation of P_E is to obtain the significant intersymbol coefficients Z_i from the generating function $H^*(z)$ calculated in Section 2.1.2. Then the integral (2.2.1) can be calculated using any one of several approaches. We used a 96 term Gauss-Legendre expansion subroutine in writing a routine which combines

a) calculation of the significant intersymbol interference coefficients $Z_i (2 \leq i \leq K)$ $\left(|Z_i| > 10^{-6}\right)$ as well as the desired signal coefficient Z_1

b) calculation of P_E using eq. (2.2.1)

The subroutine is entitled and defined as follows

PEI (DB, HNUM, IHNUM, HDEN, IHDEN, HCOMP, IHCOM, PE)

where

DB = E_b/N_o in dB

HNUM, HDEN, IHNUM, IHDEN as defined for the discrete transfer function routines (Section 2.1.2) are numerator and denominator discrete transfer function polynomials and their dimensions.

HCOMP and IHCOM are the discrete compensating filter or equalizer transfer function and its dimension (to be described in Section 2.3). The overall discrete filter transfer function is

$$\frac{\text{HNUM}(z) \cdot \text{HCOMP}(z)}{\text{HDEN}(z)}$$

P_E is the desired error probability in the presence of intersymbol interference.

2.3 Equalization or Compensation

Whether the suboptimal I&D demodulator or the optimal matched filter demodulator is used, improved performance can be achieved by additional equalization or compensation. Numerous approaches to equalizer optimization are possible, including Wiener filtering (without and with delay) to minimize the mean square error due to the combination of noise and intersymbol interference. However, since the performance criterion is error probability and the equalizer complexity (number of terms) must be constrained, we have sought to minimize the error probability for a fixed odd number N of equalizer taps by performing a minimization of the function $P_E(\underset{\sim}{h})$ over the N-1 dimensional space of variable equalizer taps. As shown in Figure 2.3.1, the middle tap gain can be normalized to unity and the remainder are varied, using a type of steepest descent numerical minimization routine to obtain the tap multiplier values which minimize P_E. *

The optimum equalizer was first obtained for N=3, with the initial parameter choice in the minimization routine being $h_1 = h_2 = 0$. The optimization was then repeated for N=5, using as the initial choice $h_1 = h_4 = 0$ with the remaining two tap parameters (around the center

*The error probability here is for uncoded operation. Tap optimization with coding is much more difficult and time-consuming. Effectively, it requires simulation for each tap value choice. However, as we found below, the optimum tap gains for uncoded operation gave comparable performance improvement with coding.

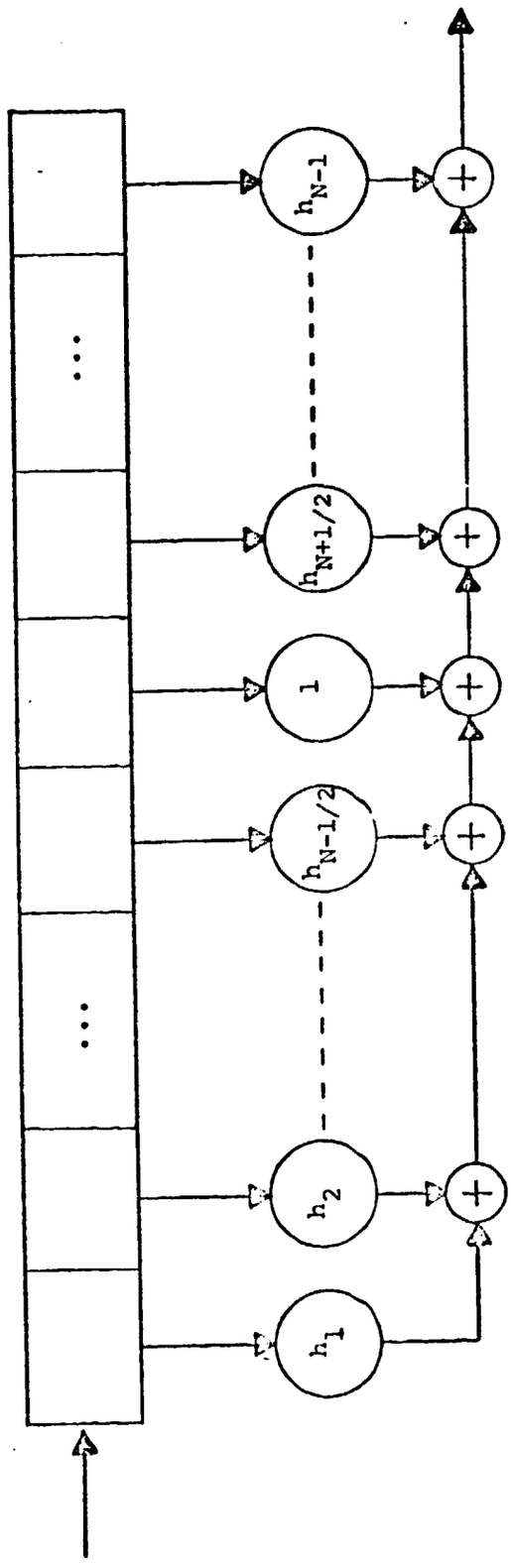


Figure 2.3.1. N stage equalizer

tap) being the minimizing parameters for N=3. This was then repeated for successive odd N until the optimum value of h_1 became negligible. In most cases, the procedure converges rapidly. For 2WT=1 a 5 tap equalizer yields most of the total performance gain achievable. In most cases, equalization produced a reduction in P_E of nearly an order of magnitude.

2.4 Performance Comparisons

As indicated in the preceding sections, the overall demodulator with equalization for intersymbol interference is as shown in Figure 2.0.1, where $G(\omega)$ is the satellite analog filter transfer function. The error probability <u>without equalization</u> for

$$\left|G(\omega)\right|^2 = \frac{1}{1+\omega^6}$$

a third-order Butterworth filter, was determined for both the I&D filter and the whitened matched filter for several 2WT products and the range of E_b/N_o of interest. Typically, the I&D filter requires 3 dB higher E_b/N_o for the same performance as the whitened matched filter at 2WT=1. Hence, we treat only the latter in detail. In Figure 2.4.1, we show for the above $G(\omega)$, the performance of the whitened matched filter, both without and with equalization for 2WT= 0.75, 1, and 1.25. We found that 7-stage, 5-stage, and 3-stage equalizers respectively yielded most of the performance improvement in these three cases. Of greatest

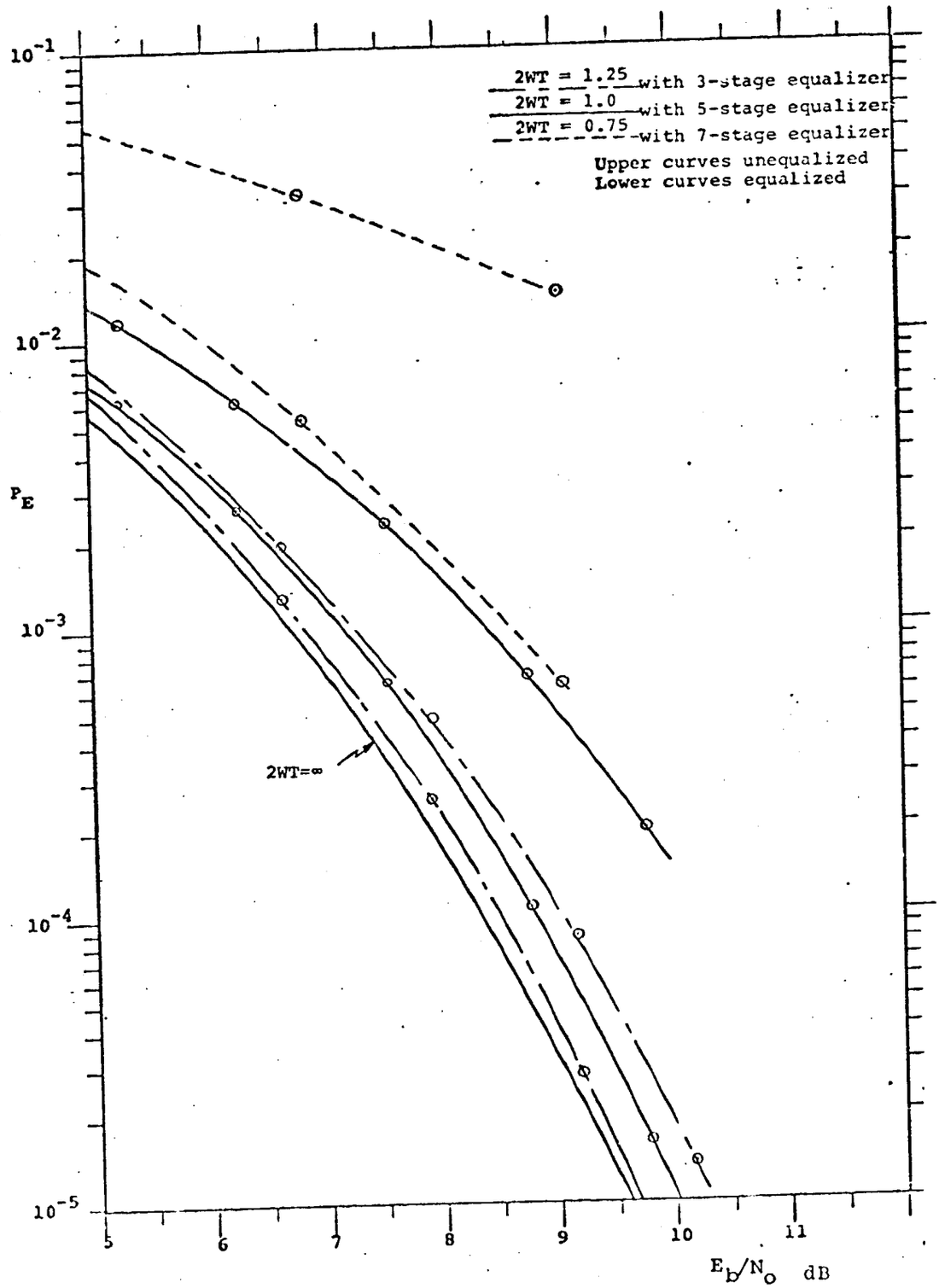


Figure 2.4.1. Error Probability for 3rd Order Butterworth Filter

significance is the fact that at 2WT=1 with a 5-stage equalizer, performance within 0.5 dB of the ideal is achieved. In Figure 2.4.2 we show for 2WT=1, and E_b/N_o = 10 dB, the performance of the whitened matched filter with and without equalization for all Butterworth filters G(ω) of order N=1 through 8. Note that as N increases, so does E_b^1, the actual transmitted energy which accounts partly for the decrease in P_E with N.

2.5 Digital Realization of Whitened Matched Filter

While it is clear that a great deal is gained by using a matched rather than an I&D filter in the presence of intersymbol interference caused by narrowband filters, the analog realization of $\overline{G(\omega)}$ even with large delay is prohibitively complex. For first-order Butterworth G(ω), the impulse response of the matched filter is a truncated growing exponential as shown in Figure 2.5.1. The discrete whitening filter is just

$$\frac{1}{H^*(z)} = c + \frac{a\, z^{-1}}{1+b\, z^{-1}}$$

which requires just one (one-bit) delay element and three tap multipliers. In fact, for an N^{th} order G(ω), the whitening filter requires N delay elements and at most 2N-1 tap multipliers.

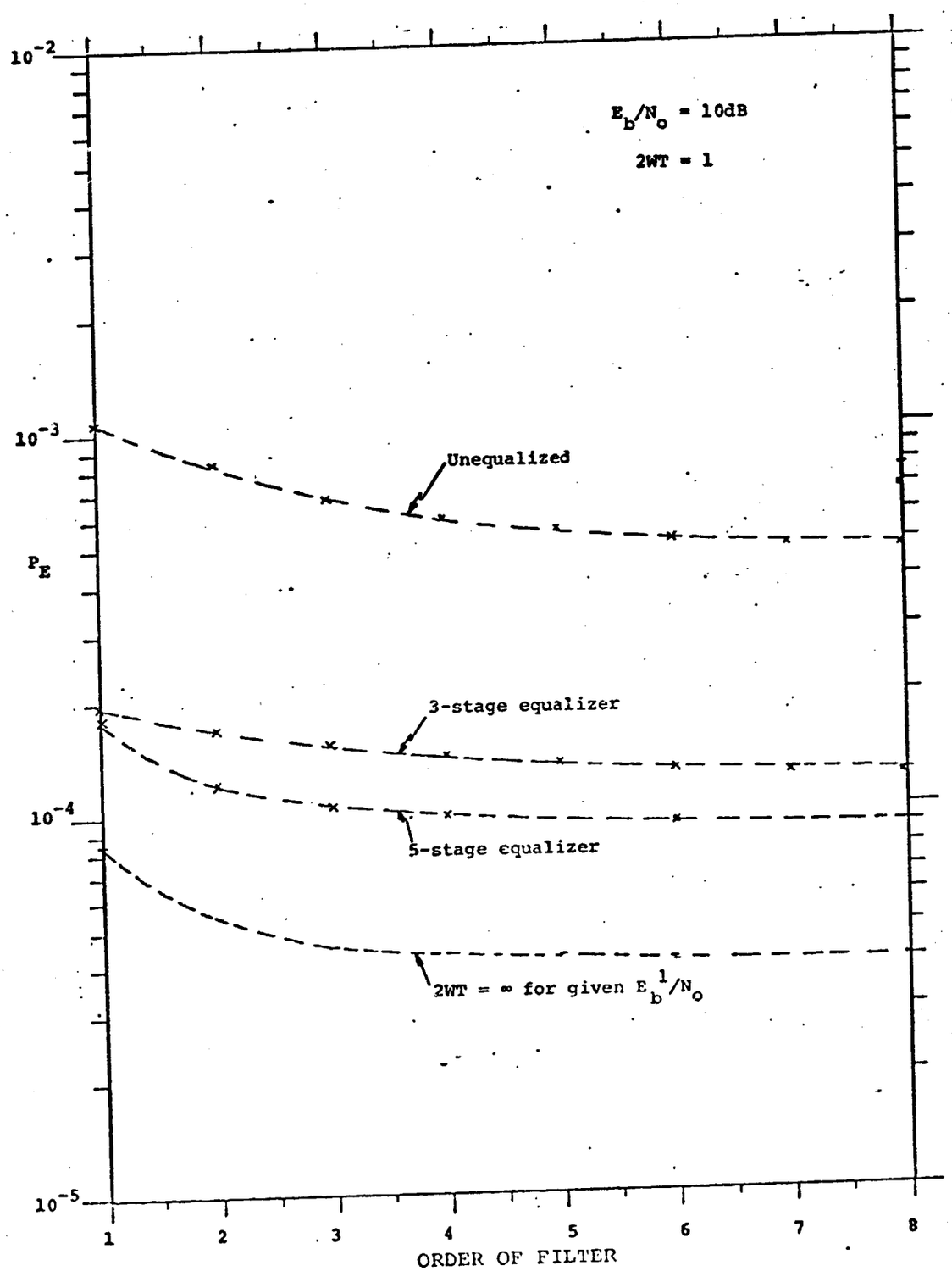


Figure 2.4.2. Error Probability for Butterworth Filter of Degree N

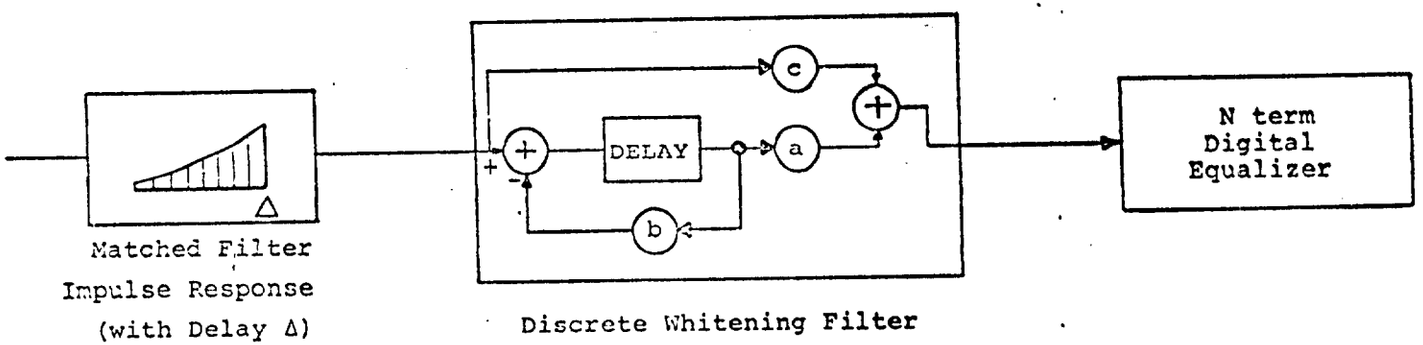
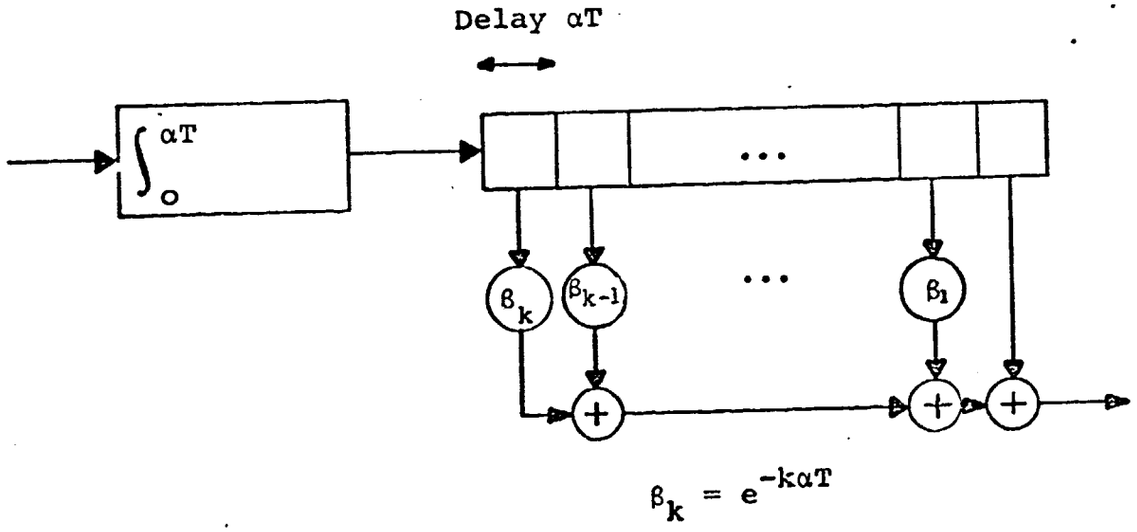


Figure 2.5.1. Whitened Matched Filter for First-Order Butterworth G(ω).

One method for approximating the matched filter digitally is to use an I&D filter which integrates over a fraction of the bit interval, αT, and to follow this by a tapped delay line as shown in Figure 2.5.2. The tap gains in this case are $\beta_k = e^{-k\alpha T}$ and K should be chosen large enough that β_K is negligible (perhaps $\leq 10^{-2}$). Also shown is the resulting impulse response of the approximate matched filter. If we take $\alpha = 0.1$, then typically K would be about 30. Also, the required quantization accuracy of this and all other digital filters must be determined. This would probably be best handled by extensive simulation.

A considerable saving in multipliers can be achieved by implementing the matched filter as a parallel combination of staggered correlators, each operating over the number of bit periods ($K\alpha$) for which the intersymbol interference is significant (Figure 2.5.3). The accumulator then sums over K integration periods ($K\alpha$ bit times) and samples at the end of this time, and the commutator alternates among the $K\alpha$ accumulators. The result is to replace the K tap gain multipliers by only $K\alpha$, although the latter must now multiply two variable numbers rather than a variable by a fixed number.



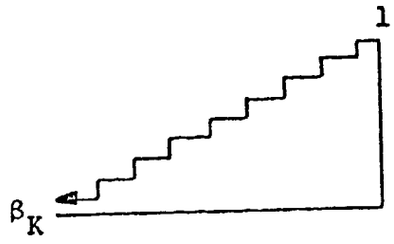


Figure 2.5.2. Digital Approximation of Matched Filter

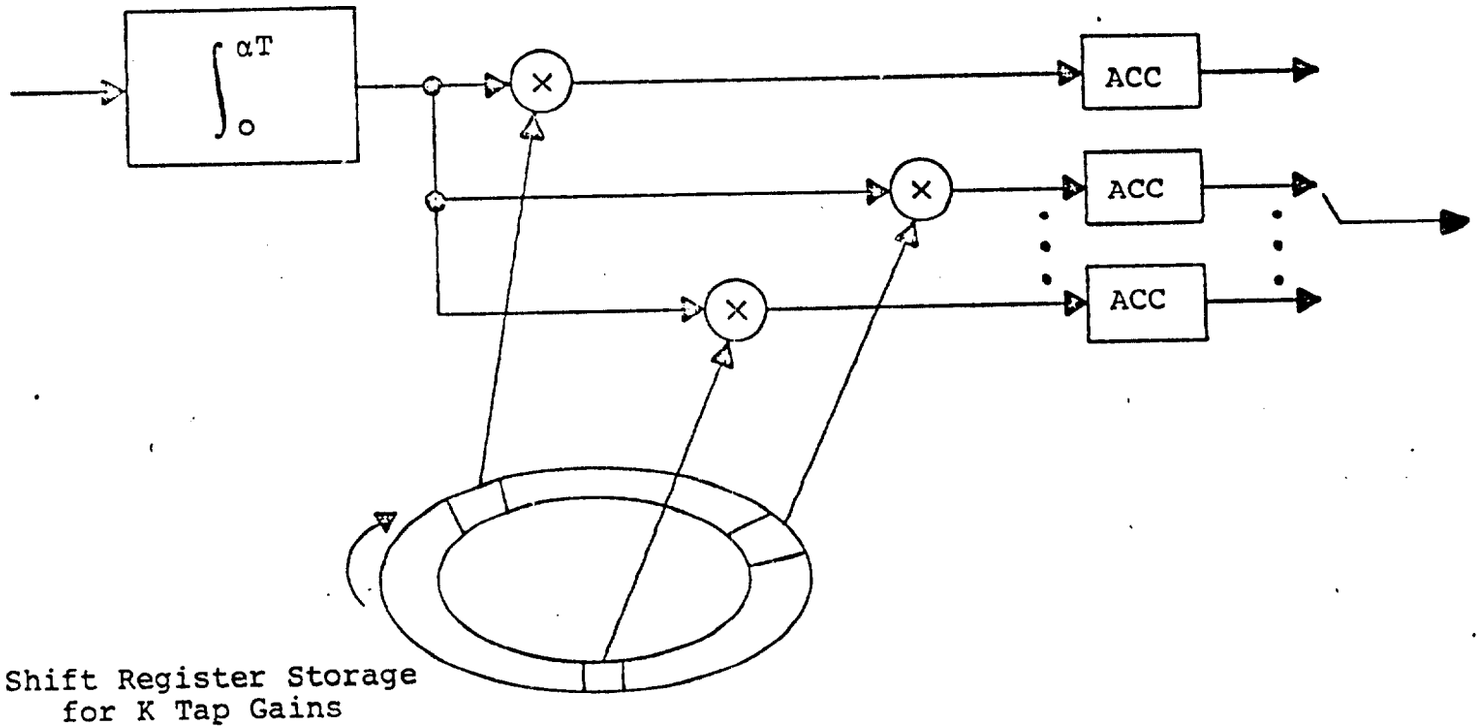


Figure 2.5.3. Correlator Approach to Digital Matched Filter Implementation.

3.0 MODULATION, DEMODULATION, AND CHANNEL CODING

The design of a modulation and channel coding system for transmitting digital video depends on the source coding method and the bandwidth and SNR requirements. To establish a benchmark for a comparison of the feasibility of several systems, assume that the video can be described with 250 Kbits/frame* (7.5 Mbps) and that the bandwidth is limited approximately to that of commercial television (4.2 MHz). The required bit error probability depends on the particular source coding method, but $P_b \leq 10^{-5}$ is sufficient for most methods.

This high data rate and restricted channel bandwidth severely restricts the modulation and channel coding techniques which are available. The following sections describe the design of signals and the selection of a coding scheme for this application. A summary of the performance of the better systems is given in Figure 3.0.1. For comparison the performance of uncoded QPSK and rate 1/2 convolutional coded QPSK systems are also given.

3.1 Signal Design for the Additive White Gaussian Noise (AWGN) Channel.

The high data rate requirement of this video system are such that with QPSK modulation the allocated channel bandwidth may be exceeded if even a small bandwidth expanding channel coding scheme is added. Yet some channel coding is useful in reducing the required satellite transmitter radiated power. A compromise can be achieved by designing signals in such a

---

* This assumes source data compression down to 1 bit/picture element.

CONVOLUTIONAL CODE	MODEM	MINIMUM CHANNEL BANDWIDTH = $1/T_c$ (T_c=Channel Symbol Time)	E_b/N_o for * $P_b = 10^{-5}$
no coding	QPSK	3.75 MHz	9.6 dB
K=8, R=2/3	Octal-PSK	3.75 MHz	6.5 dB
K=9, R=3/4	QPSK	5.0 MHz	5.5 dB
K=8, R=2/3	QPSK	5.625 MHz	5.1 dB
K=7, R=1/2	QPSK	7.5 MHz	4.5 dB

* The degradation due to intersymbol interference at the minimum bandwidth, $W = 1/T_c$, may typically be compensated by an intersymbol interference equalizer and an additional 0.5 dB of bit energy.

FIGURE 3.0.1 Summary of the Bandwidth and SNR Requirements of Several Modulation and Channel Coding Systems for a 7.5 Mbps Input Data Rate.

way as to exchange some of the coding gain for reduced bandwidth. In many cases this can be accomplished by transmitting two-dimensional (amplitude and phase) signals from an M-ary alphabet. Several different signal sets are possible such as phase-shift keying (PSK), amplitude-shift keying (ASK), and amplitude-phase keying (APK). The error probabilities of many of these configurations have been determined [1], [2]. Based on these error probabilities and the implementation complexities of the modems, we have chosen the octal-PSK modem in addition to the standard QPSK modem for further study. In Sections 3.3 and 3.4 the performance of these modems in systems with channel coding, intersymbol interference, and an intersymbol interference equalizer are discussed.

Signal design for more than two dimensions has been a topic of considerable study, but the results are almost invariably disappointing because of the enormous bandwidth expansions for classes of optimum or quasi-optimum signals (e.g., biorthogonal). A much more profitable approach is to use coding (especially convolutional) in conjunction with one- and two-dimensional signals to provide powerful multi-dimensional signals.

3.2 Channel Coding for the AWGN Channel

The state of knowledge of channel coding for the additive white Gaussian noise channel is nearly as advanced as that of modulation and signal design. As we have already mentioned in Section 3.1, block coding can be regarded as an

extension of modulating signal designs beyond two dimensions. The generally accepted constructive classes of block codes comprising the orthogonal and related code classes, fit this format exactly but are of such low rate (high bandwidth expansion) as to render them completely inappropriate here. Other constructive classes such as BCH, RS, and threshold decodable block codes, contain high rate member codes, but without exception for the given degree of complexity there exist convolutional codes which yield better performance; hence we shall concentrate on the latter.

In a rather complete study [3] of convolutional coding and decoding techniques for the AWGN channel performed for NASA Ames Research Center by LINKABIT Corporation, and in subsequent papers based on this study, ([4], [5]) it was shown that with medium code rates (1/3 and 1/2), and high data rates (1 to 10 Mbps), short constraint length convolutional codes soft-decision decoded by the maximum likelihood Viterbi algorithm are preferable to long constraint-length sequentially decoded codes, provided the bit error probability requirements are not below about 10^{-5}. Even with the sophisticated data compression techniques to be described in Section 4, it is unlikely that much lower error probabilities will be required, but in the vicinity of 10^{-8} hard decision sequentially decoded systems become competitive. For higher rate (3/4 and above) this comparison is possibly even more favorable toward maximum likelihood decoding. For example for R = 3/4, sequential

decoding requires $E_b/N_o \approx 5.8$ dB to operate at the computational cutoff rate. Viterbi decoding at this level produces $P_b \approx 10^{-6}$.

All of these studies were performed for the AWGN channel with biphase and quadriphase PSK modulation. In the following section, we extend these coding results to a system with an octal-PSK modem.

3.3 A Unity Bandwidth Expansion Modulation and Channel Coding System.

Here we describe a modulation and channel coding system which seems particularly well-suited to the bandwidth-constrained satellite channel. This readily implementable system provides moderate power gains over those required by an uncoded BPSK system with the same bandwidth requirements as that of an uncoded QPSK system. Figure 3.3.1 shows a block diagram of this system which consists of a rate 2/3 Viterbi-decoded convolutional code and an octal-PSK modem.

In the following sections, we will show that at a bit error probability of 10^{-5}, a constraint length 8 version of this system achieves a power gain of approximately 3 dB over uncoded QPSK.

3.3.1 Distance Properties

To specify the encoder and modulator of this system, the encoder tap connections and the mapping of the encoder outputs onto the signal points must be chosen. Ideally these quantities should be chosen to minimize the bit error probability. However, a computer simulation is usually required to determine

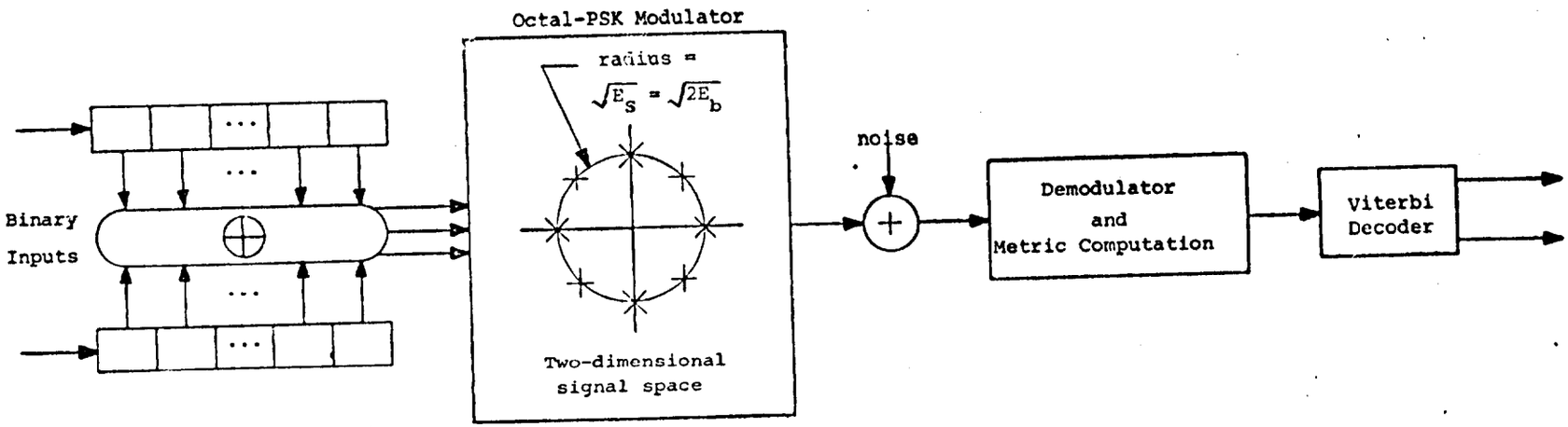


Figure 3.3.1. Block Diagram of the Unity Bandwidth Expansion Communication System.

the bit error probability. At moderate to high signal-to-noise ratios a good and usually more readily determined selection criterion is to choose the parameters to minimize the maximum word error probability between all the different pairs of code words. In additive white Gaussian noise with a one sided power spectral density of N_o the probability of decoding a particular message into another given message is

$$P_E = \int_{\frac{d}{2}}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-x^2/2} dx = Q\left(\frac{d}{2}\right) \qquad (3.3.1)$$

where

$$d^2 = \frac{2}{N_o} \sum_i x_i^2 \qquad (3.3.2)$$

and x_i is the Euclidean distance between signals in the i^{th} component. So this suboptimum selection criterion is equivalent to maximizing the minimum Euclidean distance squared between pairs of code words.

For a rate 1/v convolutional code with a BPSK or QPSK modem

$$\frac{d}{2} = \sqrt{\frac{E_b}{N_o} \frac{2k}{v}} \qquad (3.3.3)$$

where k is the number of positions in which the code words differ. So a code with the largest minimum Hamming distance between pairs of code words (d_{free}) would be chosen. The following upper bound on d_{free} has been obtained by Heller [6]

$$d_{free} \leq \min_{L = 1, 2, \ldots} \left[\frac{2^L}{2^L - 1} \frac{v}{2} (L + K - 1) \right] \quad (3.3.4)$$

where $[\ ]$ denotes the integer part of the enclosed expression.

For the system here with a rate 2/3 convolutional code and an octal-PSK modem, the best code has

$$\frac{d}{2} = \sqrt{\frac{E_b}{N_o} \mu_{free}} \quad (3.3.5)$$

where μ_{free} is the maximum over all possible codes of the minimum Euclidean distance squared between corresponding normalized (i.e., unity magnitude) components of code word pairs. The following upper bound on μ_{free}, similar to that for d_{free}, is derived in Appendix A.

$$\mu_{free} \leq \min_{L = 1, 2, \ldots} \frac{2^{2L}}{2^{2L} - 1} (L + K - 2) \quad (3.3.6)$$

Table 3.3.1 lists this bound and the corresponding bound for $2d_{free}/v$ for rate 1/2 and 1/3 convolutional codes used with a BPSK or QPSK modem.

We have chosen the encoder tap connections to produce a code with good Hamming distance properties and tried to choose the encoder output-to-modulator signal space mapping such that encoder output sequences which differ by large Hamming distances are assigned to widely separated signal points in the signal space and encoder output sequences with smaller Hamming distance separations are assigned to closer signal points. We have used a Gray code ordering for this mapping. That is, adjacent signal points are assigned to encoder output sequences,

Constraint Length K	μ_{free} upper bound for a R = 2/3 Convolutional Code with an Octal-PSK Modem	$\frac{2d_{free}}{v}$ for a R = $\frac{1}{v}$ Convolutional Code with a BPSK or QPSK Modem			
		v = 2		v = 3	
		Upper Bound	Achieved	Upper Bound	Achieved
4	4	6	6*	6.67	6.67*
5	-	8	7*	8	8*
6	6.4	9	8*	8.67	8.67*
7	-	10	10*	10	10*
8	8.53	11	10*	11.33	10.67*
9	-	12	12*	12	12*
10	10.67	13	12+	13.33	13.33*

* These represent the best $\frac{2d_{free}}{v}$ possible for noncatastrophic codes ([7], [8])

+ The best known noncatastrophic code has $d_{free} = 12$. However, a code with $d_{free} = 13$ may exist.

Table 3.3.1 Upper Bound on the Distance Parameter of Several Convolutional Coding Systems.

which differ by a Hamming distance of one, as shown in Figure 3.3.2. Unfortunately, due to the nonlinear nature of the encoder output-to-signal point mapping, it is difficult to determine μ_{free} for this code.

3.3.2 Operation and Performance

Figure 3.3.3 shows the unquantized maximum-likelihood receiver for this modulator and channel encoder. The demodulator computes the projection of the received vector on the X and Y or real and imaginary axes and these projections are used, as shown, to compute the metrics. Here metric M_i is the branch metric for a branch with a hypothesized signal phase of 45° times i. Defining r_c' and r_s' as the projections on the reference axes rotated 45° clockwise, the table lists the metrics in two equivalent ways. Without quantization both methods will, of course, yield the same results. However, with quantization, it will be shown that the obvious method, Method 1, is inferior to Method 2.

The receiver with quantization and a typical quantizer characteristic are shown in Figure 3.3.4. The projections on the reference and 45° rotated axes are computed, quantized, and used to compute the metrics as shown. To illustrate the difference between the two methods of obtaining the metrics, Figure 3.3.5 shows the M_o metric values assigned to the different regions of the received signal space for 3 levels of quantization and quantizer output values of 0, 1, and 2. That is, when the zero phase signal is hypothesized, the received signal space is partitioned and the metric values

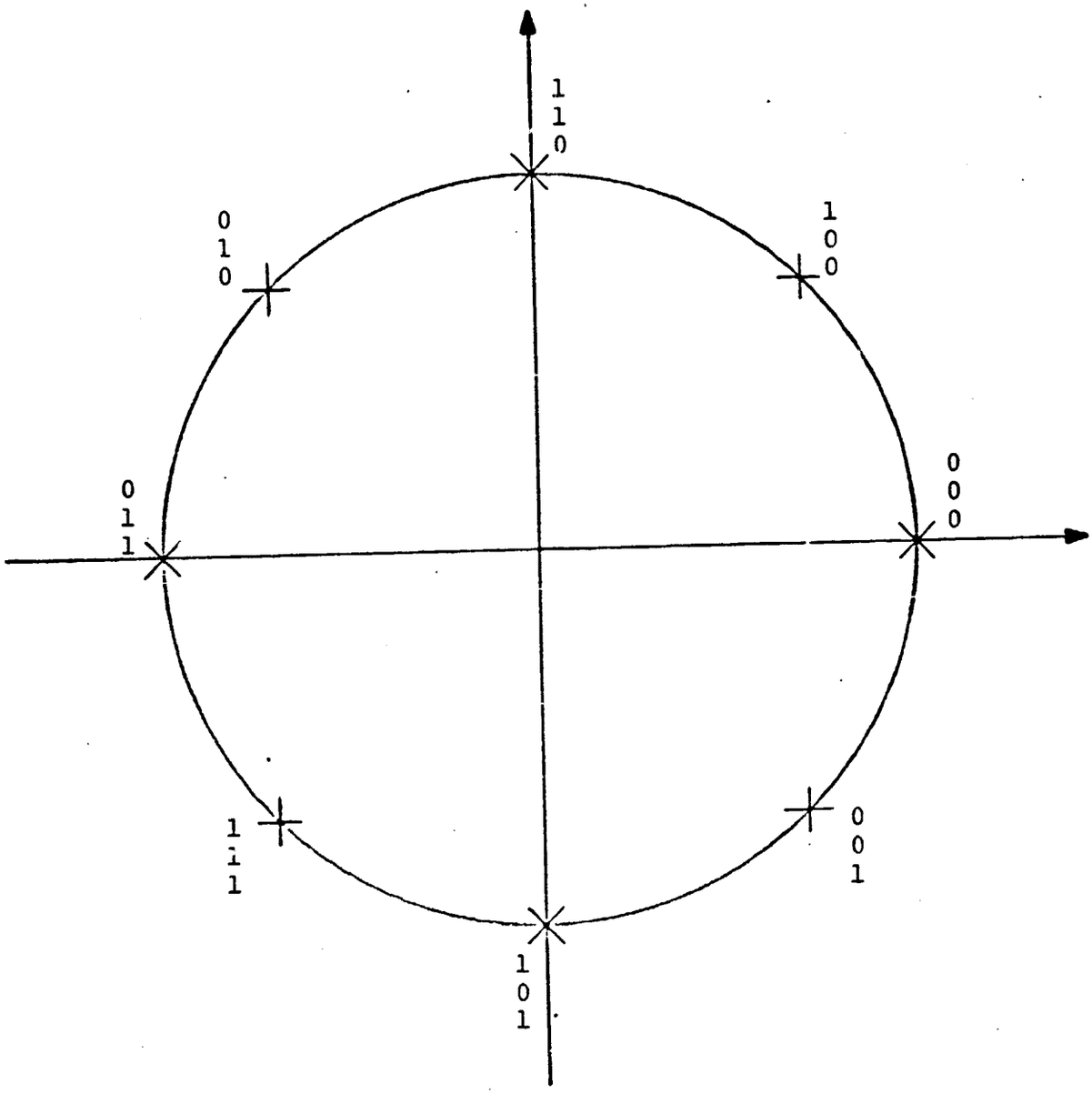
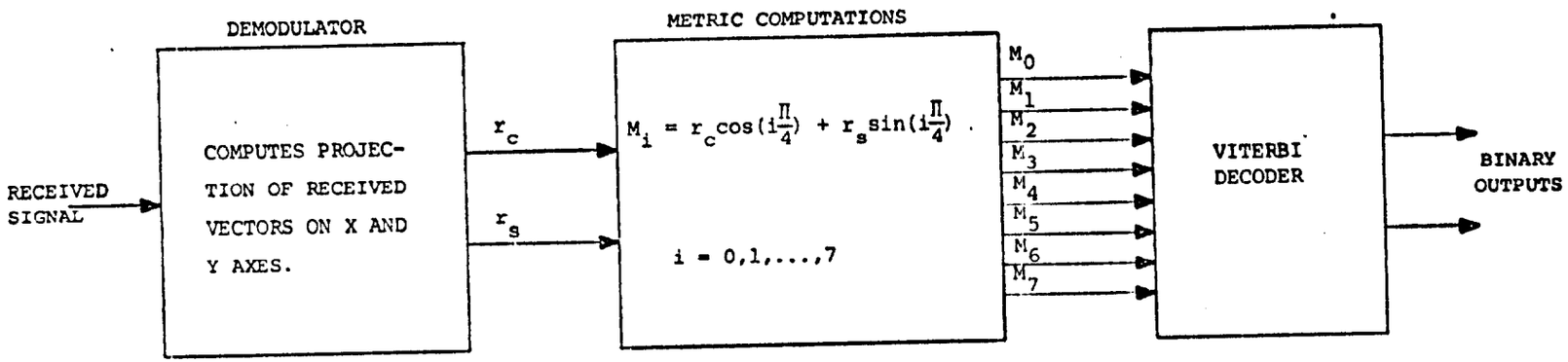
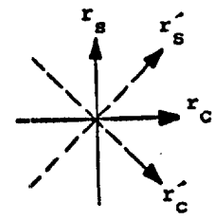


Figure 3.3.2. Encoder Output-to-Modulator Signal Point Mapping.





METRIC	METHOD 1	METHOD 2
M_0	r_c	$1/\sqrt{2}\ (r_c' + r_s')$
M_1	r_s'	$1/\sqrt{2}\ (r_c + r_s)$
M_2	r_s	$1/\sqrt{2}\ (-r_c' + r_s')$
M_3	$-r_c'$	$1/\sqrt{2}\ (-r_c + r_s)$
M_4	$-r_c$	$-1/\sqrt{2}\ (r_c' + r_s')$
M_5	$-r_s'$	$-1/\sqrt{2}\ (r_c + r_s)$
M_6	$-r_s$	$-1/\sqrt{2}\ (-r_c' + r_s')$
M_7	r_c'	$-1/\sqrt{2}\ (-r_c + r_s)$

$$r_c' = \frac{1}{\sqrt{2}}\ (r_c - r_s)$$

$$r_s' = \frac{1}{\sqrt{2}}\ (r_c + r_s)$$

Figure 3.3.3 Maximum-Likelihood Receiver with no Quantization.

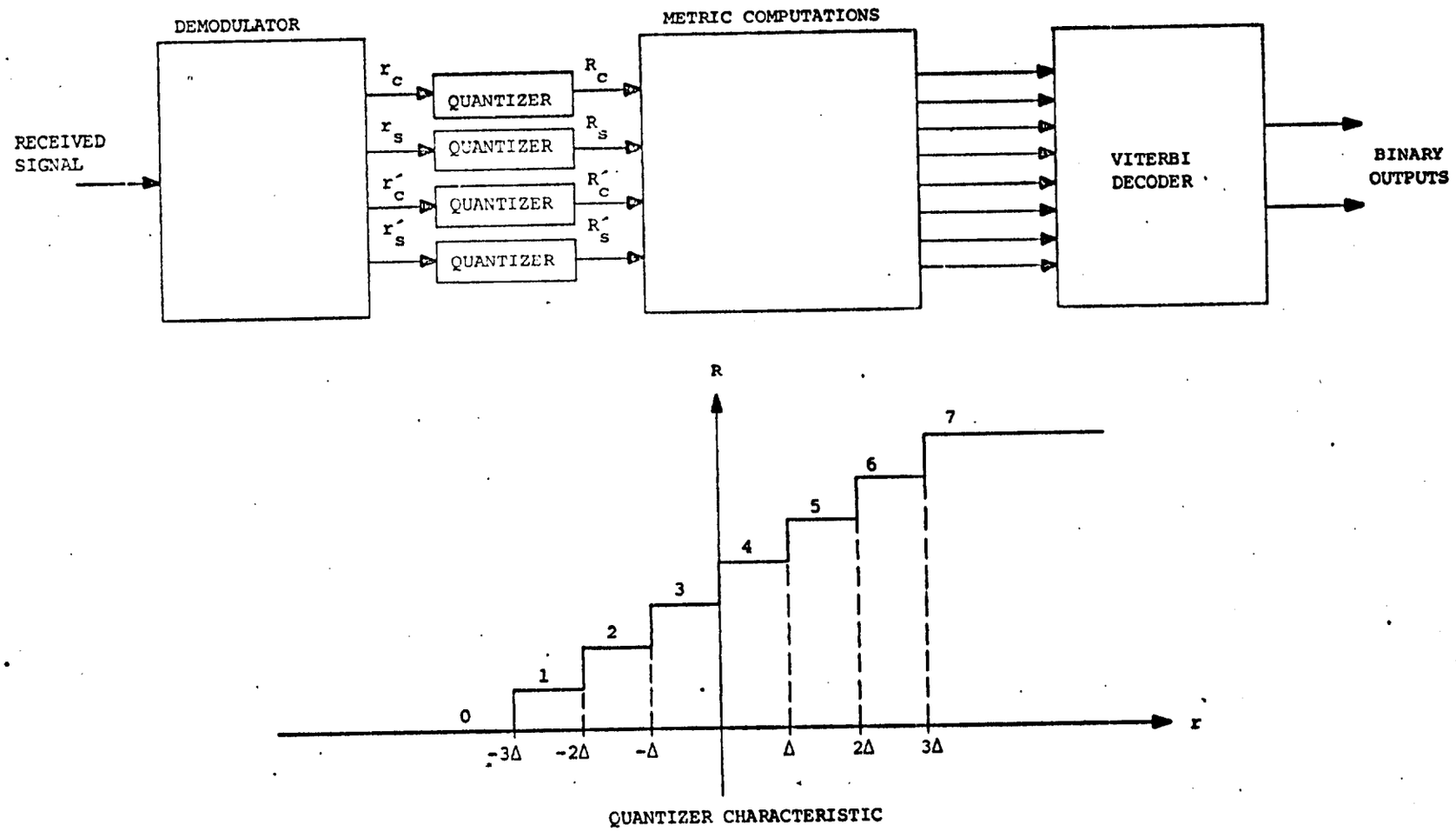
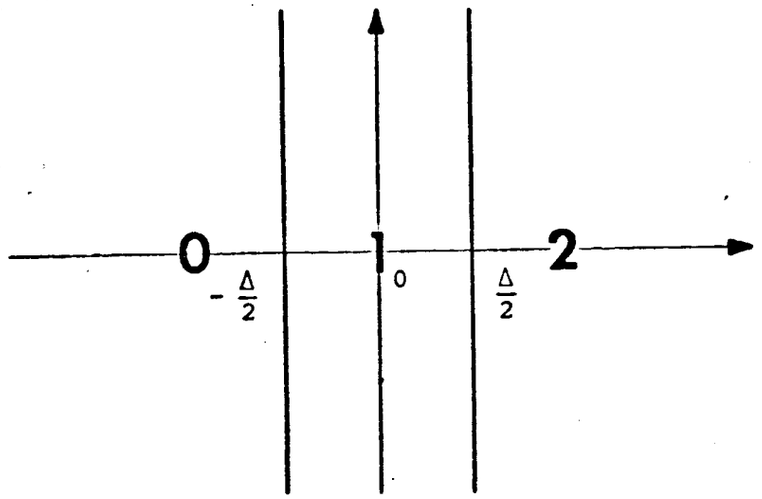
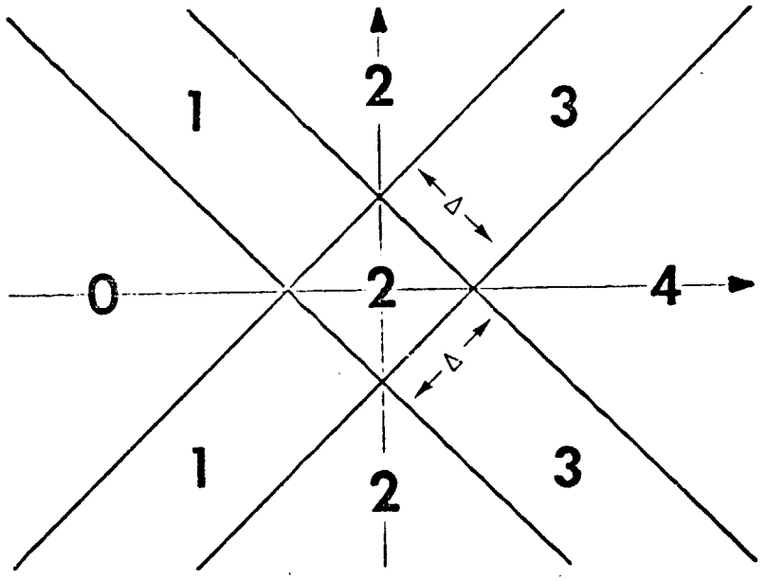


Figure 3.3.4 Receiver with Quantization.



METHOD 1



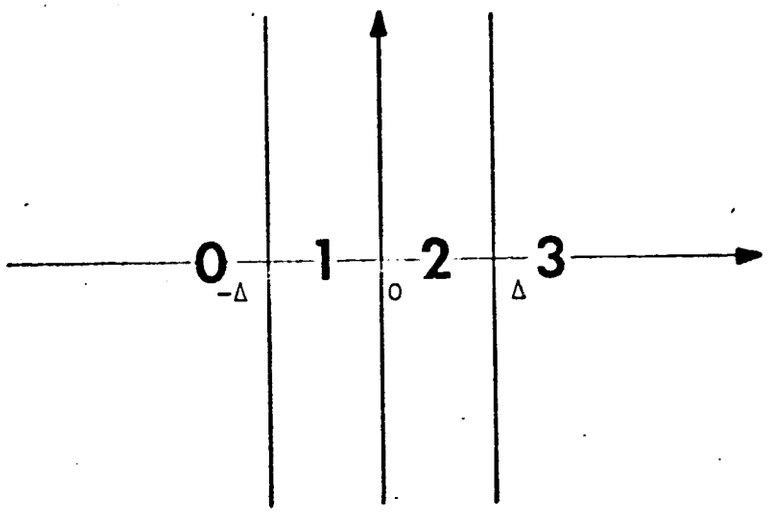
METHOD 2

Figure 3.3.5 Received Signal Space Partitioning and Metric Assignments for 3 Levels of Quantization and the Zero Phase Hypothesized Signal.

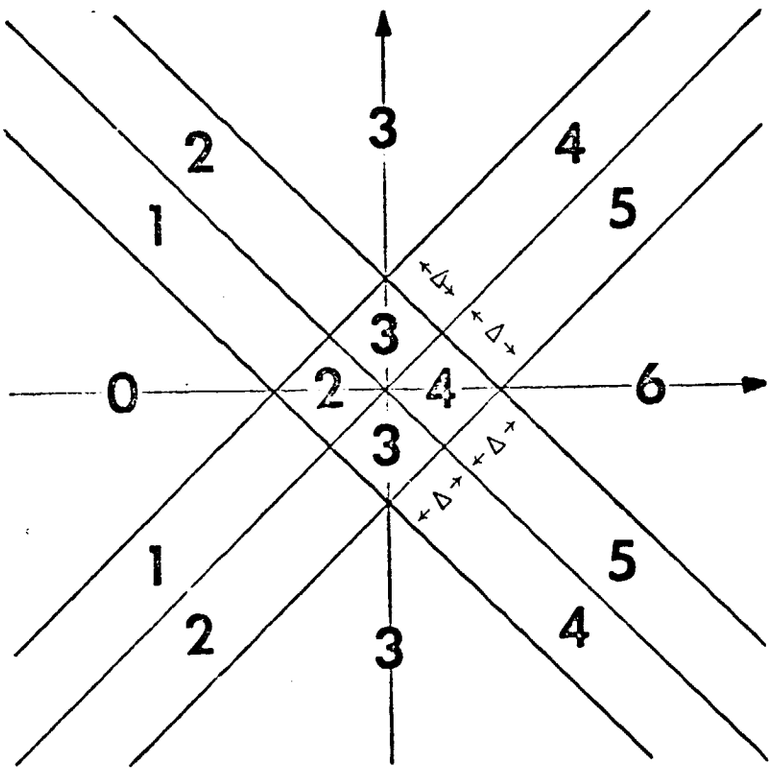
are assigned as shown. Clearly, Method 2 is better. In fact, for large signal-to-noise ratios, the M_o metric is the same for a transmitted signal with a phase of 0, +45, or -45 degrees. Figure 3.3.6 shows the corresponding received signal space partitioning and metric assignments with 4 levels of quantization. In the following sections, the second quantization method is always used.

A computer program was written to simulate the performance of this system. Figure 3.3.7 gives the results of these simulations for a constraint length 8 system with 3 bits of receiver quantization and path bit memories per state of 32, 64, and 128 bits.

This modulation and channel coding system was also simulated with intersymbol interference and the equalizer described in Section 2. Figure 3.3.8 gives a block diagram of the intersymbol interference and equalization procedures which were used in the in-phase and quadrature channel portions of this simulation. The intersymbol interference is generated by a tapped delay line approximation to a third-order Butterworth transponder IF filter and the correction is provided by a 5-stage equalizer designed for a data bit time-bandwidth product of 0.5. To account for the truncated satellite filter response the bit energy is increased by $\left(\sum_{i=1}^{9} h_i^2\right)^{-1}$. All the parameters are computed as described in Section 2. This simulation showed that in this case, the



METHOD 1



METHOD 2

Figure 3.3.6 Received Signal Space Partitioning and Metric Assignments for 4 Levels of Quantization and the Zero Phase Hypothesized Signal.

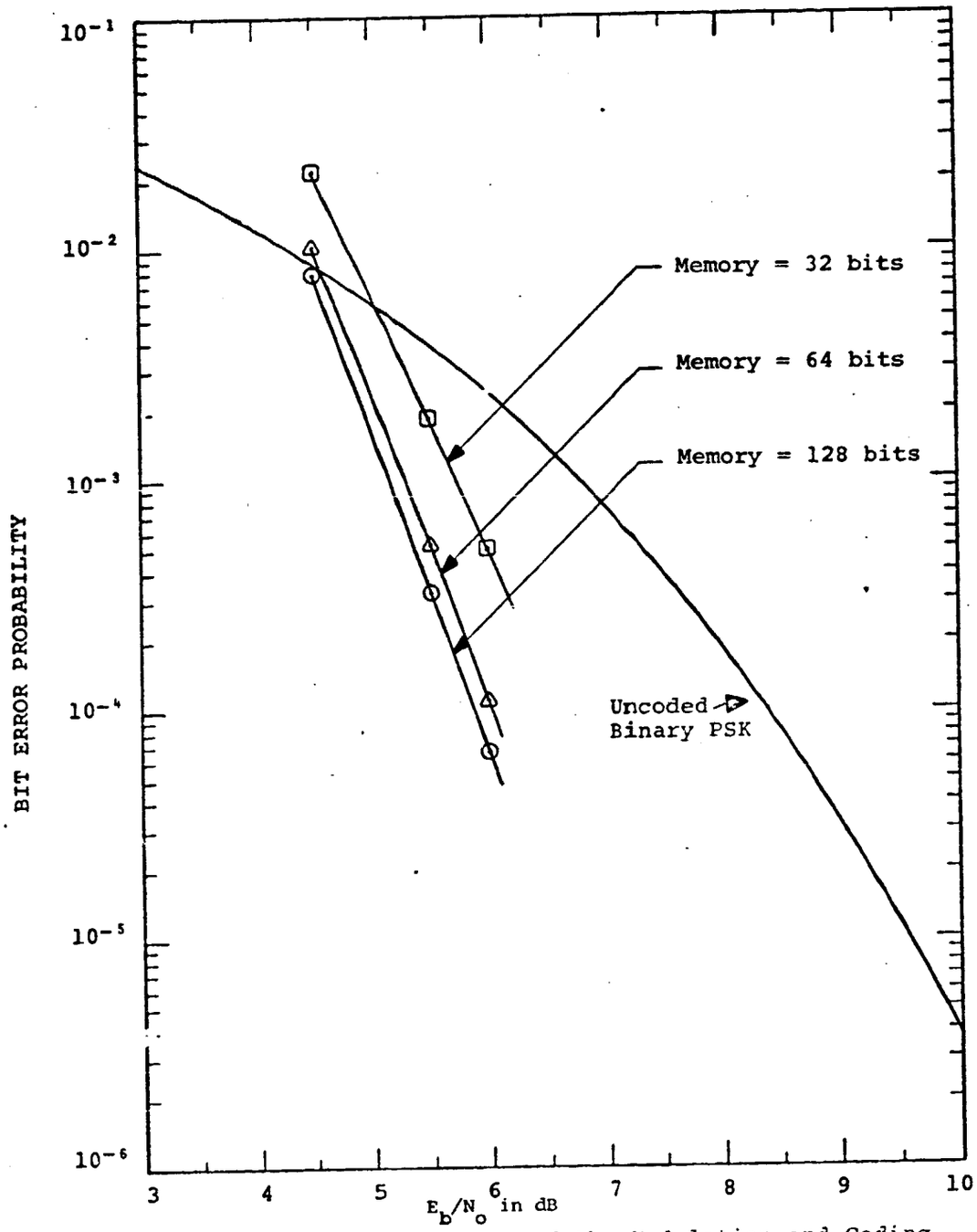


Figure 3.3.7 Performance of the Modulation and Coding System with a Rate 2/3, Constraint Length 8, Viterbi-Decoded Convolutional Code and an Octal-PSK Modem for Several Path Length Memories.

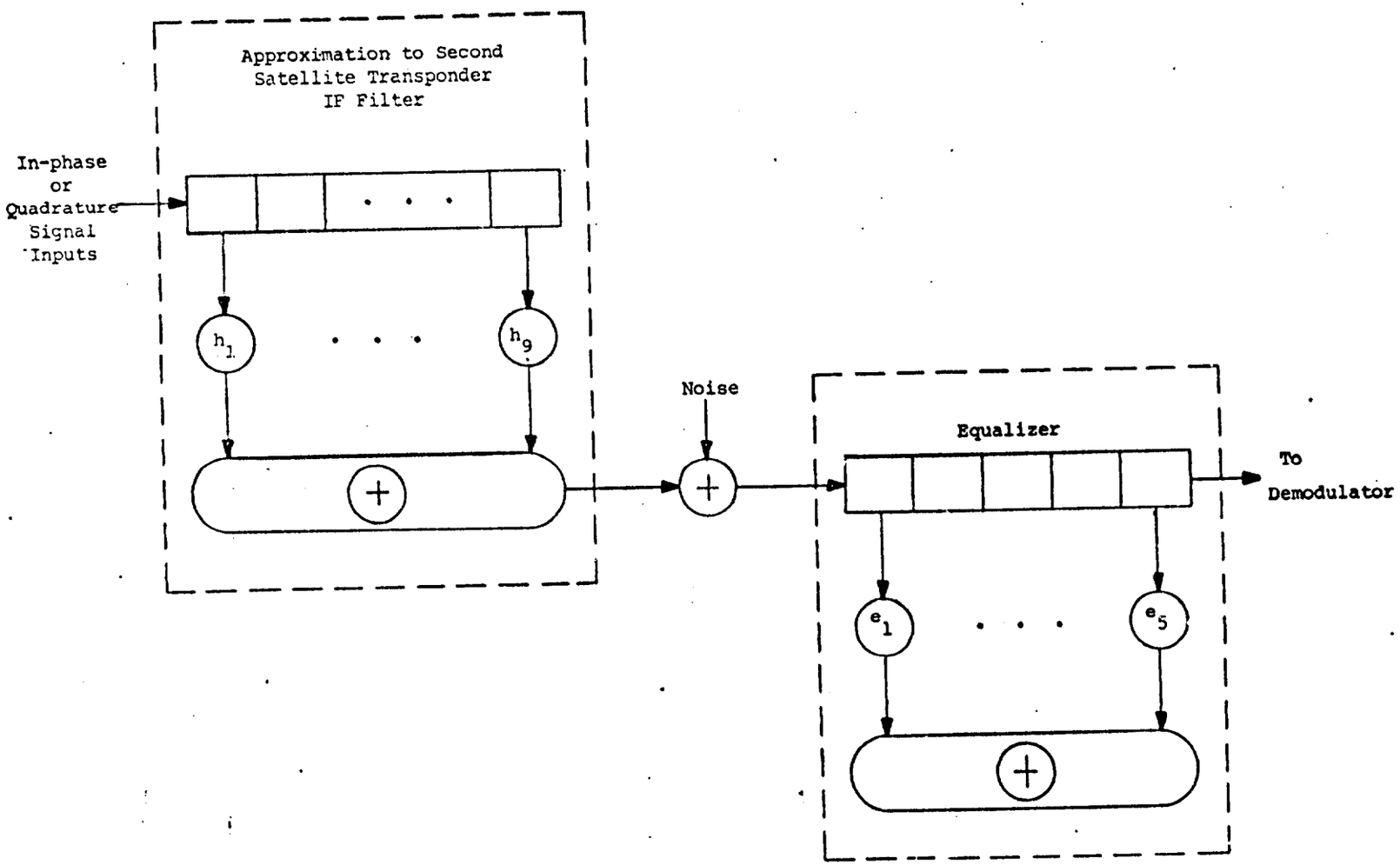


Figure 3.3.8. A Block Diagram of the Intersymbol Interference and Equalization Procedures used in the Computer Simulation.

E_b/N_o loss with respect to the same system without intersymbol interference is 0.5 dB. This is the same loss that was experienced with the corresponding uncoded BPSK system of Section 2.

3.3.3 Implementation

The major cost in implementing this unity bandwidth expansion system for the teleconferencing application would be in developing a high data rate (3.75 megasymbols per second) octal-PSK modem to provide the in-phase and quadrature signal components for the decoder.

The 7.5 Mbps decoder would require a higher speed version of the 2 Mbps K=8, R=2/3 LINKABIT LV7015MR decoder [9]. In addition, the following changes would be required.

1) Larger path memories. Systems with smaller bandwidth expansion factors require larger path memories.

2) Different branch metrics.

The metric change primarily involves storing different metrics, but the larger path memories would result in a moderate increase in decoder complexity. Using the LAM-40, LSI chip, LINKABIT is developing for the 10 Mbps, K=7, R=1/2 LINKABIT LV7017 decoder, it is estimated that the 7.5 Mpbs decoder for this system could be implemented with about 405 chips. All of the decoders mentioned here have internal capability for self-synchronization and unreliable system detection and the

chips to provide this capability are included in the above estimated chip count.

3.4 A Rate 3/4 Convolutional Coding System.

A reasonable compromise between power savings and bandwidth expansion for this application can be obtained with a K=9, R=3/4 convolutional encoder/decoder and a QPSK modem. For a 7.5 Mbps data rate this system would require a 5.0 megasymbol per second QPSK modem and an E_b/N_o of 5.5 dB to obtain a bit error probability of 10^{-5} [9]. Thus compared to the unity bandwidth expansion system of the previous section, this system provides a 1 dB power savings at the expense of a 33% larger channel bandwidth

The main implementation differences are:

1) This system uses a less complex but slightly faster modem.
2) The decoder for this system could be implemented with smaller path memories since it has a larger bandwidth expansion factor. It is estimated that the decoder for this system can be implemented with approximately 367 chips compared to the 405 chips required to implement the decoder of the previous section.

Thus, if the slightly larger bandwidth of this system is acceptable, this system provides a power and implementation savings over the unity bandwidth expansion system of the previous section.

4.0 SOURCE CODING

Source coding of the various teleconferencing sources plays a major part in compressing the amount or rate of data to be transmitted and thus reduces both bandwidth and power requirements. In this section we evaluate promising data compression techniques for several sources, beginning with those of the most demanding, i.e., the video source. We also describe the video processing system LINKABIT has developed to evaluate the various video and hard copy compression techniques.

4.1 Video Source Compression

The primary standard of comparison for video source coding schemes will be good commercial quality TV. In digital terms this corresponds to a matrix of about 500 x 500 picture elements (pels) with about 64 grey levels (6 bits) per pel (47 dB signal-to-quantization-noise). The frame repetition rate is 30 frames/sec. Thus an uncompressed PCM bit rate of about 45 Mbps is required. The objective of this video data compression study is to find practical coding techniques which yield minimum source "degradation" for a range of compression ratios. Here degradation is measured in terms of parameters related to subjective video quality.

The more successful and efficient video compression techniques developed to date have taken advantage of both statistical redundancy and human psychovisual response [10]. Spacial statistical redundancy manifests itself in terms of correlation between nearby pels within a frame and the

nonuniform first order pel grey level distribution. Temporal redundancy comes about because, although about 30 frames/second are required for apparent flicker freedom, the number of changed pels in successive frames is usually small.

There are a number of psychovisual phenomena which can be exploited to allow for source compression. For example, the eye is relatively insensitive to absolute brightness but is sensitive to relative brightness in small areas, and brightness response is approximately logarithmic with source intensity. These facts make nonuniform grey level separation effective. Greater compression ratios are attainable using techniques which exploit frame-to-frame correlation because the eye's spacial acquity drops rapidly as motion (frame-to-frame pel differences) increases. In fact a change of scene is accompanied by a loss of spacial acquity which takes several frame times to progressively reacquire [11].

Many familar performance measures such as mean square error are quite unreliable as measures of visual "quality". For example, since there is a relatively tiny proportion of high frequency "energy" in a typical scene, a minimum mean square error compression scheme might simply ignore the "highs". This, of course would cause ringing on the edges in the scene, which is a particularly objectionable form of distortion to the viewer. In this study we used criteria closely related to human psychovisual response in devising and evaluating compression techniques. This approach

sacrifices the mathematical tractability of the strictly statistical coding methods, but provides a more meaningful measure of picture quality.

4.1.1 The LINKABIT Video Processing System

To visually evaluate the various video compression techniques, LINKABIT has developed a video processing system which allows us to store and process up to 600 frames of video. This system vertically samples the horizontally scanned lines of a frame, quantizes the samples to 8 bits, and reads them into the computer where various processing schemes can be tried. After processing, the video is reconstructed by converting the sets of samples into horizontal lines of video. Since the video is sampled vertically and played back horizontally, this method distorts and rotates the picture. Presently this is corrected by sampling and reconstructing the rotated picture such that a rotation in the opposite direction of the first rotation is achieved. This double processing of the video adds some noise and distortion, but most of the degradation can be removed by carefully tuning the system.

A more detailed description of the LINKABIT video processing system follows. Figure 4.1.1 gives a block diagram of the system. The blocks in this diagram and their function and implementation complexity in terms of the number of boards and TTL chips used are:

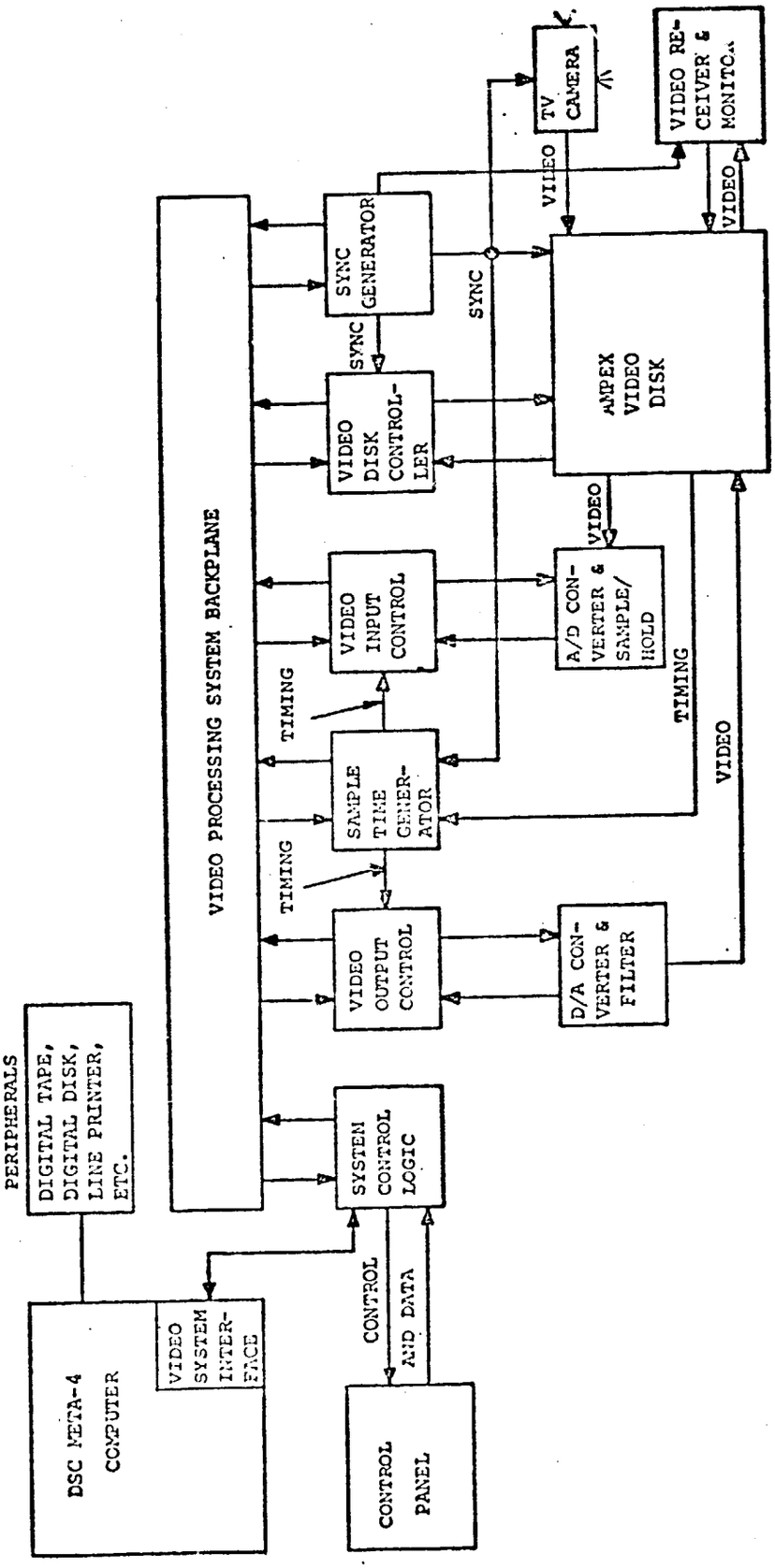


Figure 4.1.1 LINKABIT Video Processing System Block Diagram

1) The synchronization generator (1 board, 24 TTL chips). This unit provides industry compatible synchronization signals. The computer has no control over this module.

2) The video output controller (1 board, 48 TTL chips). This board contains a semiconductor memory of 512, 8-bit words to buffer a horizontal line of video from computer memory onto the video disk. In most cases 6-bit PCM samples are sufficient. However, the 8-bit words in this memory allow experimentation with up to 8 bits per sample. The video system has direct memory access to the computer allowing the memory to be filled in 2 milliseconds. The output controller logic receives a signal from the sample generator when the samples in its memory are to be dumped onto the video disk at 10^7 samples/second.

3) The sample generator (2 boards, 75 TTL chips total). The sample generator establishes sample timing for the A/D and D/A converters. Registers in the sample generator, loaded by the computer contain:

   a) The number of the first and last visible horizontal line of interest in field one and field two.

   b) The number of the next horizontal line to be written onto the video disk by the Video Output Controller.

   c) The distance from the beginning of any horizontal line of video to the first sample

to be taken by the Video Input Controller (in 100 ns increments).

d) The distance between samples taken by the Video Input Controller (in 100 ns increments).

Given signals from the video disk, the sample generator can determine the physical position of the disk at all times. Signals are created which represent the proper frame, field, horizontal line, and horizontal position of each sample.

4) The video input controller (1 board, 20 TTL chips). This unit interfaces an 8-bit A/D converter to the video system, and gives it direct memory access for data transfer. Under program control, the video input controller can operate in a variety of modes. It can transfer data in a packed or unpacked format and sample from either or both video fields.

5) The video disk controller (1 board, 42 TTL chips). The video disk controller allows the computer on-line, or the control panel off-line, to position the READ/Write heads and put the Ampex DR-10 video disk into its various modes. During normal system operation, no manual adjustment of the disk's controls is ever necessary. The controller allows each record/playback head to step over, record, playback, or erase one or several frames of video independent of the other head. This, and the fact that input and output logic is implemented separately, makes overlapped computer input and output possible. While a record/playback head is being used for output onto the video

disk, the controller allows the head to be uncommitted except during the one to four horizontal lines written during each frame time. Therefore, the system can be programmed so that a frame can be viewed on the video monitor while it is being written onto the disk. The controller includes logic implementing automatic full speed or slow motion record and playback.

6) The system control logic (1 board 49 TTL chips). This board contains logic required to operate the video system on-line from the computer or off-line from the control panel.

4.1.2 Buffer Free Techniques

Since scene activity is rarely uniform within a frame, many compression techniques generate data at a nonuniform rate, while the scene is being raster-scanned at a uniform rate. In these cases a buffer is required to smooth the data stream for transmission. We discuss here only buffer free systems where compressed data are generated at a fixed uniform rate.

Two buffer free techniques are differential and pseudo-random noise coding. Differential PCM involves the digital coding of differences between adjacent pels. A compression ratio of 2:1 on 8-bit PCM samples has been attained using this technique with nonuniform difference quantization in early

Picturephone work [12]. Differential PCM with nonuniform quantization is more effective than ordinary PCM because small pel-to-pel variations are represented accurately by fine gradations whereas large changes (edges) are much more coursely quantized. An approximately equivalent PCM system would need the finest gradation of DPCM throughout its dynamic range.

When using 2 or 3 bits of quantization per pel, the primary objectional feature observed is "false contouring" at quantum level changes. The apparent quality can be improved by adding a 3 or 4 bit pseudo-random number with peak-to-peak value, of one pel quantum level, to the pel prior to quantization. During reconstruction the PN number is subtracted off. The effect is that the quantization noise appears random rather than in concentrations which depend on the data. Such random noise is subjectively much more acceptable even though its mean square value is the same. Compression ratios of from 2 to 3 are possible here [13].

Although buffer free compression techniques allow only modest compression ratios, they are interesting because of implementational simplicity and relative insensitivity to channel errors. The latter is true because there is always a fixed number of bits generated for each pel. These bits contain information on pel amplitude only - not pel location as is the case in most buffered schemes. Channel errors appear as noisy picture elements and will not cause loss of

line or frame sync.

These techniques do not achieve a compression sufficient to digitally transmit good commercial quality television on a channel with commercial television bandwidth. However, in applications where a smaller picture (i.e., fewer horizontal scan lines per frame and/or fewer samples per scan line) or a non-real time playback are acceptable, these techniques are very attractive.

4.1.3 Polynomial Interpolators and Predictors

The so called zeroth and first order prediction and interpolator algorithms seek to fit portions of a line of pels with straight lines such that the difference between the true pel value and the interpolation or prediction line is less than some aperature value. Data, which includes pel location as well as amplitude, is generated at a non-uniform rate by these algorithms; therefore, a rate smoothing buffer is required. In addition certain controls are required to make certain that the buffer neither overflows during intense scene activity, nor underflows. This usually takes the form of variation of the error aperature width as a function of the buffer fullness.

Interpolators and predictors form piecewise linear approximations of raster scanned video signals. An interesting heuristic argument can be made as to why a piece-wise linear approximation rather than some other functional

approximation (such as a partial Fourier series expansion) seems most effective in terms of compression. The eye requires fine quantization of the low spacial frequency representing the undulating tones making up most of a typical scene. On the other hand sharp edges must also be reproduced faithfully. Piecewise linear approximation is quite effective for both of these purposes. Edges become vertical lines and undulating tones become long shallow lines. Of course, fine high frequency "texturing" is suppressed, but it is of secondary importance. The point is that we must account for psychovisual properties in selecting the "basis" for an efficient compression scheme.

Compression ratios of from about 4:1 to 2:1 are attainable on 8-bit PCM samples with predictors and interpolators [14]. These rather low compression ratios are partly due to the fact that pel amplitude and location data must be sent. Like all buffered systems, predictors and interpolators are quite sensitive to channel errors. An error in a location bit means essentially a loss of sync. Resynchronization after each line is probably necessary, and a lower channel error rate is required than in buffer free systems. The small improvement in compression for interpolators and predictors compared with some buffer free techniques does not, at this point, offset the increased complexity and synchronization requirements. Thus these techniques are not recommended for the teleconferencing application. All the one-dimensional techniques discussed thus far have been the subjects of

considerable attention and experimentation. The mediocre performance documented has led us to avoid further works in these area.

4.1.4 Two-Dimensional Methods

The compression schemes described thus far have taken advantage of statistical and psychovisual properties in only one picture dimension. We consider here techniques which are more effective because information in both spacial dimensions is accounted for in the source coding process.

One such technique is contour coding [15]. Here the high and low spacial frequencies in a frame or subframe are treated separately. The very low frequency tone information is transmitted using PCM. The PCM rate is quite low because the specially lowpass filtered scene is sampled only at the lowpass Nyquist rate. The high frequencies or edges are obtained and transmitted by following contours located by detecting connected pels with large gradients associated with them. "Synthetic highs" formed from the received contours are added to the lows to reconstruct the frame. Contour coding requires a reference memory of up to one frame. In addition, access to this memory is directed by the contour in the scene, and hence is not well structured. A buffer is also required to smooth the data generated by the algorithm. Potential implementation complexity is partially offset by the large compression possible (5:1 to 15:1 on 8-bit PCM samples). However, attainable compression is very sensitive to the number of edges in the frame and hence the type of scene. In cases where this degree of implementation complexity is acceptable, frame-to-frame compression techniques are superior.

Another class of techniques involves performing a two-dimensional transform of part or all of a frame. One such technique generates the Hadamard transform of small portions of a frame [16]. For example, consider 4 by 4 squares of pels in a frame. We may represent these 16 pels as a linear combination of 16 orthonormal 4 x 4 squares of pels. The question then is what set of orthonormal "squares" to select. One such set, suggested by the mean square error criterion, is the Karhunen-Loeve set based on the 16 x 16 correlation matrix of the pels. It turns out [16], not unexpectedly, that the correlation matrix is essentially a constant matrix with all entries very close to one. Thus, one eigenfunction (square) has a meaningfully large eigenvalue ($\sim$1) while the others are close to zero. The 16 components of the eigen "square" corresponding to the unity eigenvalue are all equal in value (i.e., this eigenvalue communicates the average grey level in the 4 x 4 square).

These results indicate that almost any set of orthogonal squares--which includes the equal component (DC) square--will be useful from a mean square error point of view. The basis should then be selected based on known psychovisual properties of the eye. One basis which is simple to implement as well as psychovisually natural is the Hadamard basis. This basis is natural because only three vectors are required to represent the great majority of effects meaningful to the eye. One vector (the DC square)

represents a constant grey level. Two other vectors represent a horizontal and vertical edge.

Compression ratios of from 3:1 to 4:1 on 8-bit PCM samples can be attained by quantizing the coefficients of these three vectors relatively finely and the remainder coarsely. The pels are reconstructed from the received basis coefficients. Even though the 4x4 Hadamard transform technique yields only a 3 or 4 to 1 compression ratio it has several advantages. First, it is a buffer free technique; secondly, it is quite naturally adaptable to compression techniques which take frame-to-frame redundancy into account. This two-dimensional technique was studied experimentally as the limiting case of the three-dimensional techniques considered below.

4.1.5 Frame-to-Frame Redundancy Reduction

In real time television applications it is also possible to exploit frame-to-frame correlation for further data compression. The major problem here is that a reference memory of at least one, and possibly many, frames worth of data must be stored in the source coding process. Although presently available IC technology makes memories of this size feasible, an important measure of practicality for any frame-to-frame technique will be the size of the encoder memory.

One approach is to transmit information only concerning those pels which have changed significantly from one frame to the next [17]. Since significant pel changes tend to occur in connection areas near the edges of a moving object,

it is more efficient to code runs of pel changes than individual changes. A smoothing buffer is necessary in this type of system since source activity is variable, depending on the amount of scene motion. It has been found that a buffer of about one frame is sufficient to yield compression ratios of from 6:1 to 8:1 with Picture-phone type scenes [17].

Another new and interesting frame-to-frame technique involves three dimensional (2 spacial, 1 temporal) transforms. For example a three-dimensional Hadamard basis might be used to represent 4 x 4 x 4 cubes of picture elements. The arguments for the efficiency of the Hadamard basis can be extended to include the time dimension. Edges of objects in motion can be approximated by planes, which will code efficiently using the Hadamard basis. An advantage of this technique is that it does not require a rate smoothing buffer. However, a reference memory of several frames is necessary.

The main disadvantage of both of these techniques is the large memory requirement. LINKABIT has devised a technique which achieves approximately the same compression as these schemes, but with a much smaller memory. This scheme begins by transmitting the compressed two-dimensional Hadamard transform coefficients for the first frame. Then for several frames quantized differences between the compressed Hadamard transform coefficients of a subpicture and the

corresponding subpicture in the past frame are transmitted. After a few frames, this procedure is repeated by again transmitting a compressed two-dimensional frame independent of previous transmissions. The additional compression over that obtained with a two-dimensional Hadamard transform scheme is possible because the difference coefficients can be quantized much more coarsely than the regular two-dimensional transform coefficients. The smaller memory is possible because it is sufficient to store a compressed rather than an uncompressed form of the reference frame.

Since this hybrid Hadamard transform technique requires a smaller memory than the other frame-to-frame compression techniques, we chose it for a more detailed study. We also performed a more detailed study of the three-dimensional Hadamard transform technique. A three-dimensional Hadamard transform video compression system is under construction at NASA Ames Research Center. This system will permit an extensive evaluation of this technique. The hybrid and three-dimensional Hadamard transform studies are described in the following sections. In both cases the techniques were tested on the LINKABIT video processing system using data sequences representative of an instructional type program with some fast action. To facilitate comparisons the center portion of the video was displayed in a split screen manner with the uncompressed, but digitized, video on the left and the compressed video on the right. Several different quantization schemes were used and the better video sequences were recorded on video tape.

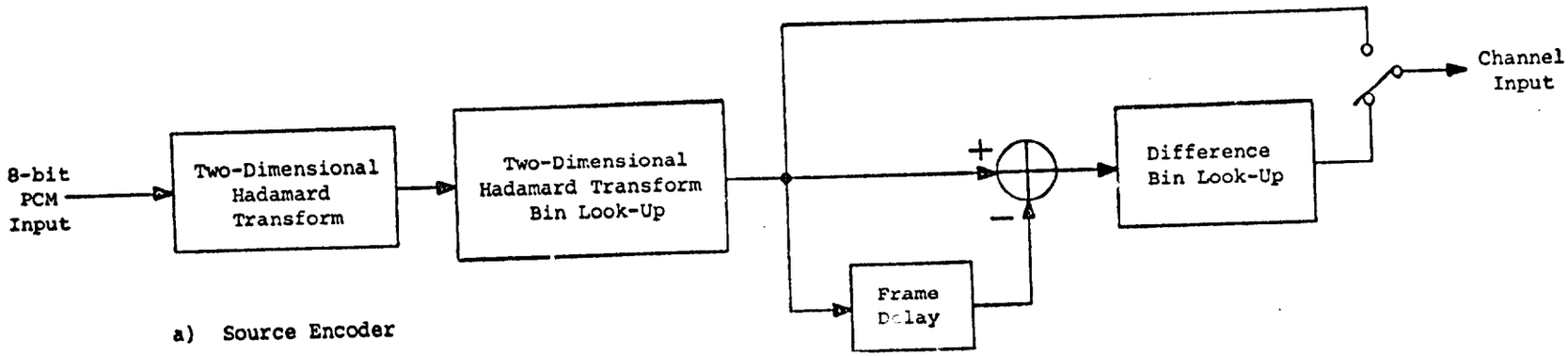
4.1.5.1 A Hybrid Hadamard Transform Compression Technique

As described in the previous section, the LINKABIT Hybrid Hadamard Transform compression technique begins by transmitting the compressed two-dimensional Hadamard transform coefficients for the first frame. Then for several frames more coarsely quantized difference coefficients, derived as shown in Figure 4.1.2, are transmitted. After transmitting several frames of difference coefficients, the picture is refreshed by again sending compressed two-dimensional Hadamard transform coefficients.

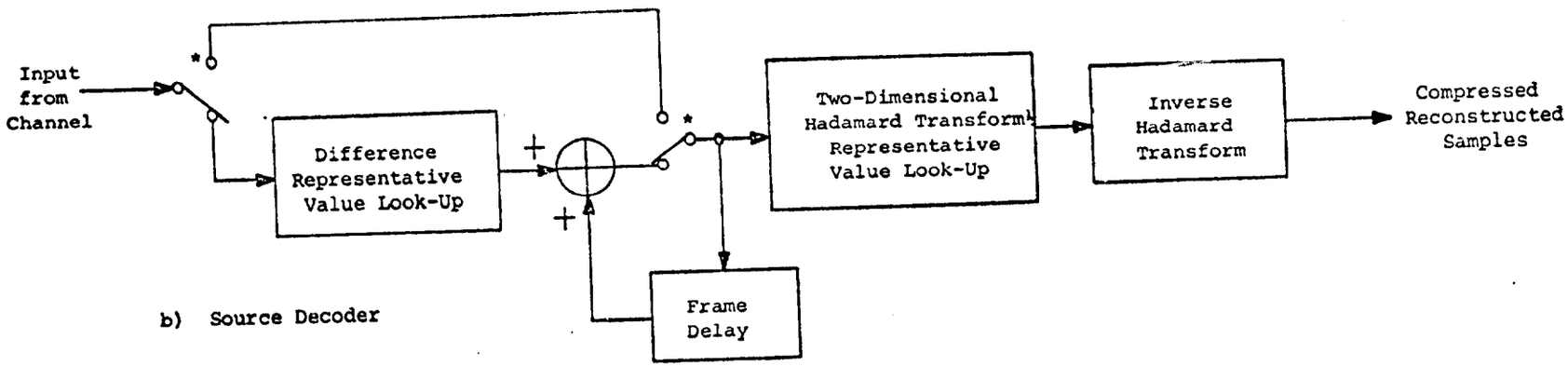
The quantizations in this scheme is performed by dividing the range of the variable to be quantized into disjoint intervals or bins, whose endpoints are called cut points, and by associating a representative value with each interval. We also assign positive integer numbers to the intervals, counting from the most negative to the most positive interval.

In testing this technique on the LINKABIT video processing system the two fields were treated as if they were processed in two separate systems. That is, the data from each field was compressed independently and then interleaved to obtain the total reconstructed picture.

Thus the two-dimensional Hadamard transform for each field is performed separately on a set of samples corresponding to a subpicture with a vertical dimension equal to twice its horizontal dimension. In active scenes where the 1/60 second field separation might product significant changes in the two field images, this method of processing the fields separately still produces a highly correlated set of samples for Hadamard transforming. This processing method is also advantageous in reducing the "false contouring" at PCM quantum level changes, which is particularly objectional in stationary scenes. For example, in a system with an input level near a quantum level of the analog-to-digital converter, contiguous subpictures from the two fields could have slightly different DC values. When these samples are interleaved the DC values alternate and an observer sees the average DC level.



a) Source Encoder



b) Source Decoder

* The switches are up for the refreshing frame and down for the remaining frame.

Figure 4.1.2 A Block Diagram of the Hybrid Hadamard Transform Compression Technique.

An alternative procedure, that deserves further attention, is to neglect the spacial differences in the two fields and to use field-to-field rather than frame-to-frame differencing. This will, undoubtedly, cause some degradation in high spacial frequencies in the vertical direction. However, due to the timing uncertainties in the system, the degradation may be small. This type of compression could be implemented with approximately half the memory of the interleaved frame-to-frame techniques.

A subpicture size of 4 x 4 samples was chosen for the tests of this technique. Studies [18] have shown that the subjective quality of n x n subpictures is essentially independent of n for $\geq$ 4. Since the implementation complexity increases with increasing n, n=4 is a reasonable choice.

The cut points and representative values of the quantizations were chosen to correspond closely to the values found by Landau and Slepian [16]. They chose their nonuniform quantization to represent the high probability regions of the variable to be quantized more accurately than the low probability regions. They also allowed more bits to represent the larger variance coefficients. Extensive testing indicated that it is unlikely that

significant improvements could be made in the subjective quality of the two-dimensional Hadamard transform scheme with different quantizations.

Figure 4.1.3 shows the 4 x 4 Hadamard set of subpictures where a white area represents plus ones and a shaded area represents minus ones. The number in parentheses above each subpicture indicates the number of bits used to transmit the coefficient of that subpicture in the two-dimensional Hadamard quantizations. For input samples between 0 and 255, the DC Hadamard transform coefficient is between 0 and 4080 and the other coefficients are between - 2040 and + 2040. We used a uniform quantization for the DC coefficient with cut points 64i, i=1,2,...,63, and representative values -32+64i, i=1,2,...,64. The cut points and representative values of the other quantizations are given in Table 4.1.1.

Several quantization schemes were used for the difference coefficients. One quantization scheme which resulted in only minor degradation in real time picture quality used 11 bits per difference subpicture per frame for an overall average of $\frac{65}{64}$ = 1.015625 bits per pel. In this case the difference coefficients corresponding to the Hadamard DC value and single vertical and single horizontal edges were quantized to 5,3, and 3 bits, respectively. Table 4.1.2 shows the groupings and the representative value assigned to each group for this quantization.

4-Bit Quantization		3-Bit Quantization		2-Bit Quantization	
Cut Points	Representative Values	Cut Points	Representative Values	Cut Points	Representative Values
421	483	140	191	63	141
309	363	62	98	0	16
215	260	16	35	- 63	- 16
138	174	0	+ 4		-141
77	105	- 16	- 4		
34	54	- 62	- 35		
9	19	-140	- 98		
0	2		-191		
- 9	- 2				
- 34	- 19				
- 77	- 54				
-138	-105				
-215	-174				
-309	-260				
-421	-363				
	-483				

Table 4.1.1 Hadamard Transform Coefficient Quantization Cut Points and Representative Values.

5-Bit Quanitzation				3-Bit Quantization
		-64 ⋮ -48	-51	
47 ⋮ 40	43	-47 ⋮ -40	-43	15, 14, 13, 12, 11, 10, 9 → 10
39 ⋮ 32	35	-39 ⋮ -32	-35	
31, 30, 29, 28	29	-31, -30, -29, -28	-29	8, 7, 6 → 7
27, 26, 25, 24	25	-27, -26, -25, -24	-25	5, 4, 3 → 4
23, 22, 21	22	-23, -22, -21	-22	2, 1 → 1
20, 19, 18	19	-20, -19, -18	-19	0 → 0
17, 16, 15	16	-17, -16, -15	-16	-1, -2 → -1
14, 13, 12	13	-14, -13, -12	-13	-3, -4, -5 → -4
11, 10	10	-11, -10	-10	-6, -7, -8, -9, -10, -11, -12, -13, -14, -15 → -7
9, 8	8	-9, -8	-8	
7, 6	6	-7, -6	-6	
5, 4	4	-5, -4	-4	
3	3	-3	-3	
2	2	-2	-2	
1	1	-1	-1	
0	0			

Table 4.1.2 Difference Coefficient Quantization Groupings and Representative Value Assignments.

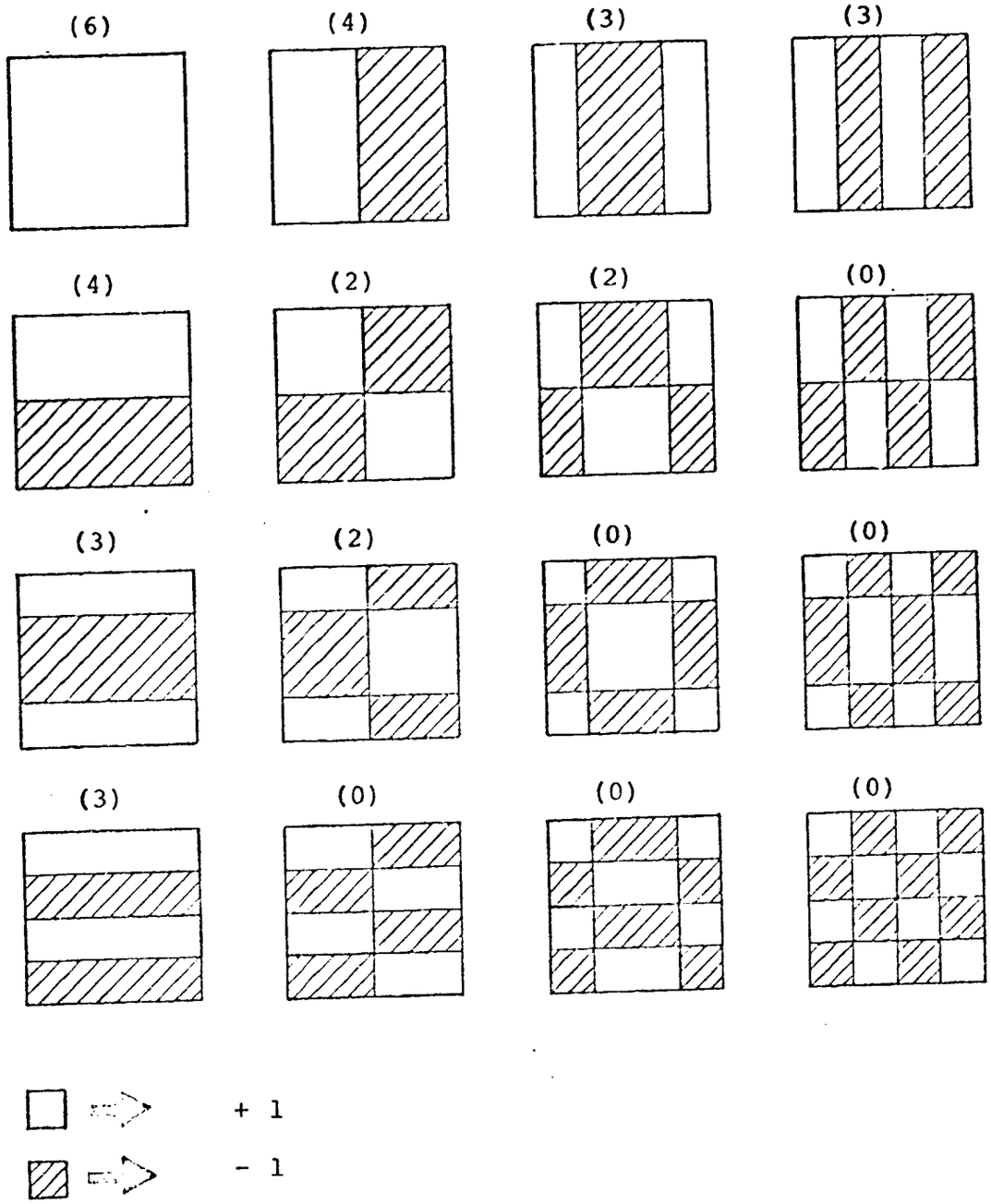


Figure 4.1.3 Hadamard Set of Subpictures

As noted, this technique and particular quantization produces only a small degradation in real time picture quality. However, in slow motion some trailing is visible in scenes with fast motion. This effect could be reduced by refreshing every 3 or 2 frames instead of every 4 frames. With the same quantization these more frequent refreshings would increase the average number of bits per pel to 1.125 or 1.34375, respectively, from the present 1.015625. Some improvement may also be possible without increasing the average number of bits per pel by using a different quantization scheme for the difference coefficients. The performance of the 1.015625 bit per pel case has been recorded on video tape.

The main advantage of this technique is its relative implementation simplicity compared to that of other frame-to-frame techniques. Instead of requiring a memory for one or more frames of uncompressed data, this technique only requires that the source encoder store one frame of compressed data. Using a 500 x 500 matrix of samples per frame, the 2 bit per pel Hadamard coefficient quantization used here requires a 0.5 M bit memory. This memory could be implemented with 128,4096-bit random-access memories (RAM) chips.

In cases where there is very little motion, this system could be designed to detect this fact and adjust the quantization to provide a finer quantization for the

refreshing frame at the expense of a coarser quantization of the difference frames. This modification would be very useful, for example, when the video is focused on a detailed scene on a blackboard or piece of paper.

The Hadamard transform operations for a system with 6-bit input samples, which is sufficient for most applications, can be implemented by adding or subtracting the incoming samples from 16 parallel accumulators corresponding to the 16 Hadamard transform coefficients. After each set of 4 samples on a horizontal line is accumulated, the result is stored while the next set of 4 samples is accumulated. Truncating the accumulated sums at 8 bits, the 16 add or subtract circuits require approximately 32 chips, the 16, 128-sample storage registers approximately 16 chips, the 16 accumulators about 16 chips, and 32 chips are required to multiplex the accumulator or storage register output to the add or subtract inputs; thus exclusing the timing requirements, a total of approximately 96 chips are required.

The Hadamard transform operations can also be implemented with a fast Hadamard transform procedure. However, this procedure appears to be more complex to implement than the procedure described in the preceding paragraph.

As with most frame-to-frame techniques this hybrid technique requires a channel rate buffer. The size of this buffer can be greatly reduced by modifying this technique to refresh 1/4 of the picture every frame instead of the entire frame every 5th frame. Using this procedure and processing 4 sample lines of a field at a time, this rate buffer requires about 8 K bits of storage. Although this method is easier to implement, it has not yet been simulated.

The nonuniform quantizations can be implemented with table look-up operations using read-only memories (ROM's)

Adding a few chips for timing and the remaining operations, it is estimated that the encoder and decoder for the compression scheme could each be implemented with approximately 350 chips. If the picture degradation of the field-to-field technique mentioned previously is acceptable, a 250 K bit reference memory would be sufficient. In this case, the encoder and decoder could each be implemented with approximately 286 chips. Thus the size and power requirements of the compressor are quite reasonable. However, the implementation requires many expensive chips.

4.1.5.2 The Three-Dimensional Hadamard Transform Compression Technique

Based on the efficiency of the two-dimensional Hadamard transform compression technique, a logical approach is to extend this technique to three-dimensions. The main problem with this is the large reference memory required. However, because of recent interest we have tested such a technique on the LINKABIT video processing system.

For these tests, we used 4 x 4 x 4 cubes of picture elements, and, as with the hybrid Hadamard transform scheme, processed each field separately.

To describe the three-dimensional Hadamard subpictures, imagine giving depth to the two-dimensional subpictures of Figure 4.1.3. If (+) denotes the polarity (i.e. shaded or not shaded) of the subpictures of Figure 4.1.3, and (-) denotes the opposite polarity, then there are four depth or time dimension cases to consider. First, the polarity could remain the same for the entire depth. We represent this as ++++ and label the coefficients $\{f_{ij1}\}$ where i and j specify the row and column, respectively, in the subpictures of Figure 4.1.3. The second case is when the cubes start as shown in Figure 4.1.3 but then alternate polarities through the depth of the cube. This is presented as +-+- and the coefficients are labeled $\{f_{ij2}\}$. The depth polarities of these and the remaining cases are shown in Table 4.1.3.

Table 4.1.3 also shows the number of bits used to transmit the three-dimensional Hadamard transform coefficients. The reasoning behind this selection is that the coefficients corresponding to the subpictures most noticeable to the human eye should be quantized finer than those corresponding to subpictures representing chaotic or rapidly changing

Depth Polarity of Three-Dimensional Hadamard Subpictures	Coefficients	Number of Bits Used To Transmit the Coefficients
+ + + +	$\{f_{ij1}\}$	$\begin{bmatrix} 6 & 4 & 3 & 3 \\ 4 & 2 & 2 & 0 \\ 3 & 2 & 0 & 0 \\ 3 & 0 & 0 & 0 \end{bmatrix}$
+ - + -	$\{f_{ij2}\}$	$\begin{bmatrix} 4 & 2 & 0 & 0 \\ 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$
+ + - -	$\{f_{ij3}\}$	$\begin{bmatrix} 4 & 3 & 2 & 1 \\ 3 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$
+ - - +	$\{f_{ij4}\}$	$\begin{bmatrix} 4 & 2 & 0 & 0 \\ 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$

Table 4.1.3. Three-Dimensional Hadamard Transform Quantization Details.

scenes on which observers can't focus their eyes or attention anyway. As shown in Table 4.1.3, we represented the coefficients $\{f_{ij1}\}$ corresponding to subpictures with no depth polarity change with 32 bits; the coefficients $\{f_{ij3}\}$ corresponding to subpictures with one depth polarity change with 16 bits; and the two sets of coefficients corresponding to rapidly changing scenes of two or three depth polarity changes with only 8 bits each. This produces an average of 1.0 bit per pel.

Since the range of the 4 x 4 x 4 Hadamard transform coefficients is four times that of the 4 x 4 coefficients, the quantization cut points and representative values for the 2-, 3-, 4-, and 6- bit quantizations where chosen to be four times larger than those given for the hybrid Hadamard transform technique of the previous section. The 1-bit quantizor had a cut point of 0 and representative values of $\pm$ 80.

With this set of parameters the three-dimensional Hadamard transform scheme produced poorer quality video than the hybrid Hadamard transform scheme on the LINKABIT video processing system. A loss of detail at edges, especially edges of moving objects, was evident. Different quantizations may improve the performance but time did not allow for the extensive experimentation required for optimzation. A real time three-dimensional Hadamard transform system is being developed by NASA Ames Research Center to provide a long term base for experimentation and optimization of such a system.

4.1.6 Color Considerations

The two color difference signals used in commercially compatible color television are used to quadrature amplitude modulate a 3.58 MHz subcarrier which is added to the base-band black and white signal prior to frequency translation and transmission. It is clear at the outset that significant compression should not be attempted on this composite signal but on the individual components themselves.

The eye is quite sensitive to color accuracy but is not very sensitive to high spacial frequency color information as long as the monochromatic edge information is present. In other words, smearing of color changes across edges is tolerated by the eye. This means that, in the 4 x 4 Hadamard square technique only the DC color term need be transmitted with fidelity. Thus, considerably less information can be sent concerning the color difference signals than the monochromatic signal. Additional compression can be obtained by using frame-to-frame correlations just as in the black and white signal case.

The LINKABIT video processing system is presently not equipped to handle color images. However, studies ([18], [19]) have shown that using two-dimensional compression techniques, transmission of the color components will require no more than one additional bit per pel.

It is estimated that using the LINKABIT hybrid frame-to-frame compression technique, the color components can be transmitted with an average of only 0.3 bits per pel. This estimate is based on human psychovisual response to the color signals and the fact that the in-phase and quadrature chrominance signals only require bandwidths of 1.5 MHz and 0.5 MHz, respectively, whereas the luminance signal requires a 4.2 MHz bandwidth. For the in-phase chrominance component, the Hadamard transform coefficients corresponding to the DC value and single horizontal and vertical edges would be transmitted with 6, 3, and 3 bits, respectively, and the DC value coefficient would be updated once with 3 bits. For the quadrature chrominance component, only a 4-bit quantized representation of the Hadamard transform coefficient corresponding to the DC value would be transmitted and there would be no updating. Thus 19 bits would be required to transmit the information contained in a 64 pel cube.

4.2 Voice Compression

There are several well developed speech compression techniques which achieve significant compression ratios. One of the most common methods is the channel vocoder, which is capable of sending acceptable speech at 2400 bits/sec, and somewhat less acceptable speech at lower rates [20]. The channel vocoder consists primarily of a bank of contiguous bandpass filters across the voice frequency band. Due to human vocal tract constraints, the envelope of the outputs of these filters varies quite slowly (<25Hz). Thus, sampling and quantizing the channel amplitudes at a low rate provides significant compression.

Further compression is possible because the outputs of the bandpass filters are highly correlated. In fact, vowel sounds are composed essentially of only 2 or 3 frequency components. Some further compression of the outputs of the channelized speech has been attempted [21, 22], with only moderate success, using very simple schemes. Many of the more complex techniques, such as defining a moderate number of equivalence classes of patterns of filter outputs and transmitting these coded patterns, are

now practical to implement. However, since the amount of voice data compared to video data will be small, these more complex techniques are not justified for this application.

4.3 Hard Copy Compression

Many compression techniques for typewritten and graphical data have been proposed and developed. One of the simpler and more effective is run length encoding. Here the length of runs of black or white data are encoded and sent. Depending upon the source statistics, rather large compression ratios are possible. Run length encoding has the disadvantage that it requires a rate smoothing buffer and it is very susceptible to sync loss due to channel errors.

Another technique devised by J.P.M. Schalkwijk [23] (see [24] for the more general formulation) is based on indexing the subset of sequences with a particular property and transmitting this index rather than the actual sequence. For example, the input sequence can be prescanned to determine the Hamming weight and this weight and the index, or order of the particular sequence among all sequences of the given weight, can be transmitted. These properties uniquely specify a fixed length input sequence.

In particular for an input sequence

$$\underline{x} = (x_1 \; x_2 \; \cdots \; x_n), \qquad x_i \in \{0,1\},$$

with weight W, the transmitted index is

$$i_s = \sum_{j=1}^{n} x_j N_j$$

where

$$N_j = \binom{n-j}{W - \sum_{k=1}^{j-1} x_k}$$

$$N_1 = \binom{n-1}{W}, \qquad W \neq n$$

and by definition

$$\binom{n}{w} = 0 \qquad \text{for } w > n$$

The successive components x_k, k=1,2, ..., n, of a data sequence of weight W can be recovered from the transmitted index by successively comparing i_s with N_k and modifying i_s as follows. If $i_s \geq N_k$, set $x_k=1$ and $i_s=i_s-N_k$; otherwise set $x_k=0$.

This method works well for small weight input sequences, but it can produce a data expansion, rather than compression, with input sequences containing nearly half ones. However, if the source can be assumed to generate independent binary digits, very little compression is possible, in this case anyway. So when the sequence has more than W_{max} (perhaps $W_{max} = .1n$) ones, the data sequence can be transmitted unchanged.

For a binary source generating independent digits with the probability of a one being p, the mean number of channel bits per source bit required by this algorithm is

$$\frac{\bar{k}}{n} = \frac{1}{n}\left[\log_2 (W_{max}+ 2)\right]^{+} + \Pr\left\{ \text{more than } W_{max} \text{ ones} \right\}$$

$$+\frac{1}{n} \sum_{i=1}^{W_{max}} \Pr\left\{ i \text{ ones} \right\}\left[\log_2 \binom{n}{i}\right]^{+}$$

$$= \frac{1}{n}\left[\log_2 (W_{max} + 2)\right]^+ + \sum_{i=W_{max}+1}^{n} p^i (1-p)^{n-i} \binom{n}{i}$$

$$+\frac{1}{n} \sum_{i=1}^{W_{max}} p^i (1-p)^{n-i}\left[\binom{n}{i} \log_2 \binom{n}{i}\right]^+$$

where $\lceil x \rceil^+$ denotes the smallest integer greater than or equal to x.

Another variation of this compression technique would be to prescan a sequence until a set weight is accumulated and then to transmit the sequence length and the index. LINKABIT has investigated the implementation complexity of this and the preceding technique based on the Schalkwijk algorithm and has determined that they can be implemented with add, subtract, and compare logic and reasonable size memories. For n=100 and W_{max}=10 the memory is less than 700 bits.

5.0 SYSTEM CONSIDERATIONS

A wide range of teleconferencing applications are anticipated. These include communication between regional networks of governmental agencies, medical hospitals, schools, libraries, computer networks, and data banks and educational broadcasting with limited report-back needs. The main factors affecting the information transfer optimization are:

1) the number of simultaneous and/or sequential accesses to other stations required;
2) the type, quantity, and fidelity requirements of the several classes of data;
3) the utility versus cost tradeoffs for the ground stations and end-user land communications;
4) the existing user satisfaction and unfilled needs in non-satellite based telecommunication networks.

Due to the wide variety of teleconferencing requirements and the fact that conclusive information is not yet available on the requirements for the various applications, system flexibility is a major consideration.

5.1 Multipoint Teleconferencing

Multipoint teleconferencing will involve situations in which each station wishes to receive data and/or video from a small number of the other points simultaneously and be able

to rapidly switch between points. Also, the traffic will vary markedly between different branches of the network. For example, station A may be broadcasting video and data to stations B, C, and D, utilizing almost the full capability of a satellite transponder. Then, A may cease broadcasting data and station B might use the resulting capacity to begin transmitting video, simultaneous with A's transmission, while station C sends hard copy to A and D. Fixed channel assignments are clearly too restrictive in a dynamic, flexible network.

The multipoint network has the following properties:

1) each station simultaneously accesses one or more other stations;
2) traffic flows change markedly in short periods of time;
3) capacity is insufficient to provide every network pair with a fixed channel assignment equal to the maximum traffic requirement between that pair.

Thus, a flexible multiple access technique providing demand assignment is required. TDMA [26-29], FDMA, code division multiple access, and orthogonal convolutional code multiple access [30] techniques are possible. A long frame TDMA system in which all source coding, multiplexing, and communication functions are performed with clocks synchronous with satellite time (to eliminate the need for bit stuffing seems to provide the most economic, flexible system for the following reasons:

1) TDMA makes the most efficient use of the limited transponder bandwidth.

2) No power control is required, and the transponder can be operated in a saturated mode, yielding maximum gain and EIRP and minimum interchannel interference, assuming amplitude modulation is not being used.

3) A station can increase its outgoing traffic by simply increasing its burst length within the frame, when capacity permits.

4) A station can receive from several other stations (have several accesses) without requiring several complete receivers, since a single receiver and decoder can be time-shared among the accesses, with phase tracking and bit-timing accomplished digitally in a single shared random access memory and arithmetic unit.

5) In some cases, the compression and expansion buffers required for TDMA can also serve as source coding buffers, particularly in frame-to-frame video data compression techniques.

6) Timing can be established at the time of network formation and maintained by the use of short preambles preceding each burst. If it is necessary to provide for new stations

to join an existing network, timing acquisition can be achieved by use of a LINKABIT scheme known as wide-aperature acquisition, in which a several millisecond aperature is left empty, perhaps one each second. A new station transmits into this aperature, which is long enough to absorb the initial timing uncertainty, listens to himself, and then nulls out the timing uncertainty.

5.2 Formatting

Formatting of the various data sources is readily established in the TDMA system by using frames or superframes for data framing. Preambles preceding each burst can be used to transmit service information. In an FDMA system, frames must be established for the data even though not needed for multiplexing channels.

Within each channel, there would be video, voice, color information, hard copy, and computer data, some of which would be in compressed form.

This data must be highly protected, since highly compressed source data is quite susceptible to errors. Also, even with a low error probability, say 10^{-5}, the source decoder must be designed to react to errors in a noncatastrophic manner. The computer-to-computer data can be more heavily protected with interleaved forward-error-correcting outer

codes. For example, a convolutional encoder with the relatively simple (35 chip) LINKABIT LF33 rate 3/4 feedback decoder [9] reduces the bit error probability from p to approximately 30 p^2 with an E_b/N_o increase of only 1.25 dB. An even better performance can be obtained by concatenating a Reed-Solomon outer code [31]. However, this concatenated scheme is typically much more complex to implement. Another possibility in cases where retransmission is feasible, is to use an error detection and repeat request (ARQ) system to obtain a very low probability of an undetected error. However, due to the additional buffer requirements, an ARQ system does not appear to be justified in this application.

5.3 Synchronization

Bit-timing and phase-tracking loops for BPSK and QPSK systems are now well understood, and are directly applicable in an FDMA system. Extensions to octal-phase systems can be made. LINKABIT has invented several techniques for minimizing loop cost in a shared receiver TDMA system. All loops are maintained coherent frame-to-frame with carrier phase and time phase information about different bursts stored digitally.

Phase and node-synchronization ambiguities can be resolved in a TDMA system by use of the frame preamble. Frame synchronization may be maintained by a distributed frame synchronization word, with 1 or 2 bits of the word appearing in each preamble.

Node synchronization may be determined in an FDMA system by monitoring the performance of the channel decoder, and selecting an alternate position if the data is bad over a protracted period. This approach is now standard in LINKABIT decoders. With a rate 3/4 code, node synchronization requires checking four different branch phases. Phase ambiquities of 180° can be resolved by use of a transparent code and differential encoding-decoding outside of the channel encoder-decoder. The situation is only slightly more complicated in an 8-phase modulation system; again, monitoring of the decoder performance can be used to resolve combined phase and node ambiguity.

The source coder must provide synchronization signals for initial acquisition plus information for maintaining synchronization despite occasional errors. In a TDMA system, synchronization of source data will be tied to the TDMA frame and superframe by appropriate formatting.

6.0 Conclusions

As noted in the introduction, key cost factors for digital teleconferencing via satellite include signal-to-noise ratio, bandwidth, complexity, and system sensitivity to channel disturbances. The required signal-to-noise ratio (SNR) is the product of two factors, E_b/N_o and source data rate. These two factors were treated separately in Sections 3.0 and 4.0, and we review them briefly.

In Figure 3.0.1, bandwidth and SNR requirements were summarized for several modulation/channel coding systems. Three combinations appear to be performance and cost effective, namely QPSK-rate 1/2, QPSK-rate 3/4, and octal PSK-rate 2/3. In Table 6.1, we reproduce the E_b/N_o and channel bandwidth requirements, and add estimates of relative complexity and relative sensitivity to channel degradations (robustness). The relative complexity numbers are obtained by assuming roughly equal cost for the modem and decoder for the rate 1/2 system and then estimating chip count and expense for each for the other two systems. The source rates are assumed to fall between 5 and 10 Mbps for the two systems.

The K=9, rate 3/4 system appears to provide a good compromise in E_b/N_o requirements and bandwidth occupancy. If

Table 6.1. Coding-Modulation System Parameters

Code	Modem	Bandwidth Factor (bits/symbol)	E_b/N_o (dB) ($I_b=10^{-5}$)	Relative Complexity	Robustness
K=7, R=1/2	QPSK	1	4.5	1	Good
K=9, R=3/4	QPSK	1.5	5.5	1.3	Good
K=8, R=2/3	OCTAL-PSK	2.0	6.5	1.7	Fair

bandwidth is tightly constrained or a significant operating expense, however, the initially more expensive K=8, rate 2/3 octal-PSK system may be a better choice. On the other hand, if powerful source coding reduces the need for tight bandwidth control and/or if E_b/N_o requirements are critical, the rate 1/2, QPSK system might be the choice. Clearly the system designer has several well-defined choices here.

Determination of actual bandwidth requirements for a given rate depends on the specific transponder characteristics and the decision as to whether or not to use a powerful equalizer. From Section 2.0, Figure 2.4.1, we note a rough rule of thumb that a bandwidth equal to the QPSK symbol rate permits operation with approximately 0.3 dB degradation with equalization. A bandwidth 25% greater is required without equalization. It is probably reasonable to assume an additional 15-25% bandwidth allowance for system non-idealities. It is estimated that the equalizer would add approximately 10% to the cost of the modem/channel decoder and would increase sensitivity to channel variations.

In Section 4.0, several techniques for video data compression were considered. Three promising schemes are summarized in Table 6.2. Scheme A, utilizing 2-dimensional Hadamard transforms with fine time differencing, has been extensively simulated and appears to be most promising, representing a good compromise among compression, cost,

Table 6.2. Characteristics of Candidate Video Compression Schemes

Scheme	Approximate Compression		(4) Bits/Pel	Quality	Estimated Cost	Sensitivity	Data Rate			
							Black & White		Color (1)	
	2D Spatial	Time					Total	Non-Video	Total	Non-Video
A	3:1	2:1	65/64	Good	1.0	Good	8 Mbps	(2) 380 Kbps	10.4 Mbps	(2) 494 Kbps
B	3:1	3:1	11/16	Fair	1.0	Good	5.5 Mbps	(2) 260 Kbps	7.1 Mbps	(2) 336 Kbps
C	3:1	3:1	2/3	(1) Good	2.0	Fair	5.0 Mbps	(3) 0	6.5 Mbps	(3) 0

A - 2-Dimensional Hadamard with fine time differences

B - 2-Dimensional Hadamard with coarse time differences

C - 2-Dimensional Hadamard with rate-buffered time differences

(1) Estimates - Simulation results not available

(2) Other data transmitted during vertical retrace sync time

(3) Buffered, so no vertical retrace time set aside for other data

(4) Add 19/64 bits/pel for color

quality, and sensitivity. Somewhat cruder time differencing permits further data reductions, but at an appreciable cost in quality. It does appear that time redundancy can be more effectively reduced without quality sacrifice if substantial rate buffering (probably equal to one or more fields of compressed data) is provided to achieve compression when only parts of the picture are changing. However, no LINKABIT simulation results are presently available for this approach. Further work in indicated.

Bandwidth requirements may be inferred from Tables 6.1 and 6.2. With compression scheme C, black and white pictures, equalization, and rate 2/3 -octal PSK modulation-coding, the bandwidth is approximately 1.0 x 5.0 Mbps ÷ 2 bits/symbol or 2.5 MHz, which appears to be the minimum for acceptable quality at reasonable signal-to-noise ratios. Here, the power signal-to-noise ratio is 6.5 dB + 0.3 dB (bandwidth degradation) + 67 dB (data rate) or about 74 dB exclusive of implementation losses and margins.

More reasonably, scheme A and rate-3/4 QPSK modulation-coding requires a bandwidth of 8 ÷ 1.5 = 5.3 MHz for black and white or 6.9 MHz for color with power signal-to-noise ratio of 77 dB and 78 dB respectively. These results are based on the adequacy of a bit error rate of 10^{-5}.

This error rate appears to be acceptable for video since, in scheme A, the picture is completely updated each 4 frames. With a bit error rate of 10^{-5}, an error event can be expected once each 4×10^{5} bits, or about once each 2 frames. The average duration is 2 frames, so it appears that some group of 16 pels would be in error on every frame. This appears to be acceptable but deserves further attention.

Error rates for non-video data should be controlled by use of a concatenated coding scheme, since it appears that sufficient data rate is available during vertical retrace for schemes A and B for expected volumes of non-video data.

Scheme C, utilizing a rate buffer to achieve further compression, is interesting in that the rate buffer can also serve as the TDMA burst buffer. Of course, in a non rate-buffered scheme, the TDMA compression and expansion buffer can be substantially smaller. If a teleconferencing network is in use, with several accesses provided for in the TDMA format, further flexibility is possible in that the burst used for each transmission might be adjusted to the data generated, permitting a reduced rate buffer. Further work is again indicated.

Appendix A Distance Properties of a Unity Bandwidth Expansion Coding and Modulation System.

This Appendix contains a derivation of the upper bound on μ_{free} of (3.3.6) for the system consisting of a rate 2/3 convolutional code and an octal - PSK modem. μ_{free} is defined as the maximum over all possible codes of the minimum Euclidean distance squared between corresponding normalized (i.e., unity magnitude) components of code word pairs. For reference the bound is

$$\mu_{free} \leq \min_{L=1,2,\ldots} \frac{2^{2L}}{2^{2L}-1} (L + K - 2) \qquad (A.1)$$

Following Reference 25 (p311), let the code contain M code words $c^{(1)}, c^{(2)}, \ldots, c^{(M)}$

where

$$c^{(i)} = \left[c_0^{(i)} \; c_1^{(i)} \ldots c_{n-1}^{(i)}\right]$$

Then the total sum of the distances squared between pairs of code words is

$$\mu_{total} = \sum_{i=1}^{M} \sum_{j=1}^{M} \mu\left(C^{(i)}, C^{(j)}\right)$$

$$= \sum_{i=1}^{M} \sum_{j=1}^{M} \sum_{m=0}^{n-1} \mu\left(c_m^{(i)}, c_m^{(j)}\right) \qquad \text{(A.2)}$$

where μ (x,y) denotes the sum of the normalized distances squared between the components of x and y. By the law of cosines these component distances squared are

$$(\text{distance})^2 = 2 - 2 \cos \phi \qquad \text{(A.3)}$$

where ϕ is the angle between the components in the signal space of Figure A.1.

For any code let

$$N^{(m)} = \left[N_1^{(m)} \quad N_2^{(m)} \quad \ldots \quad N_8^{(m)} \right]$$

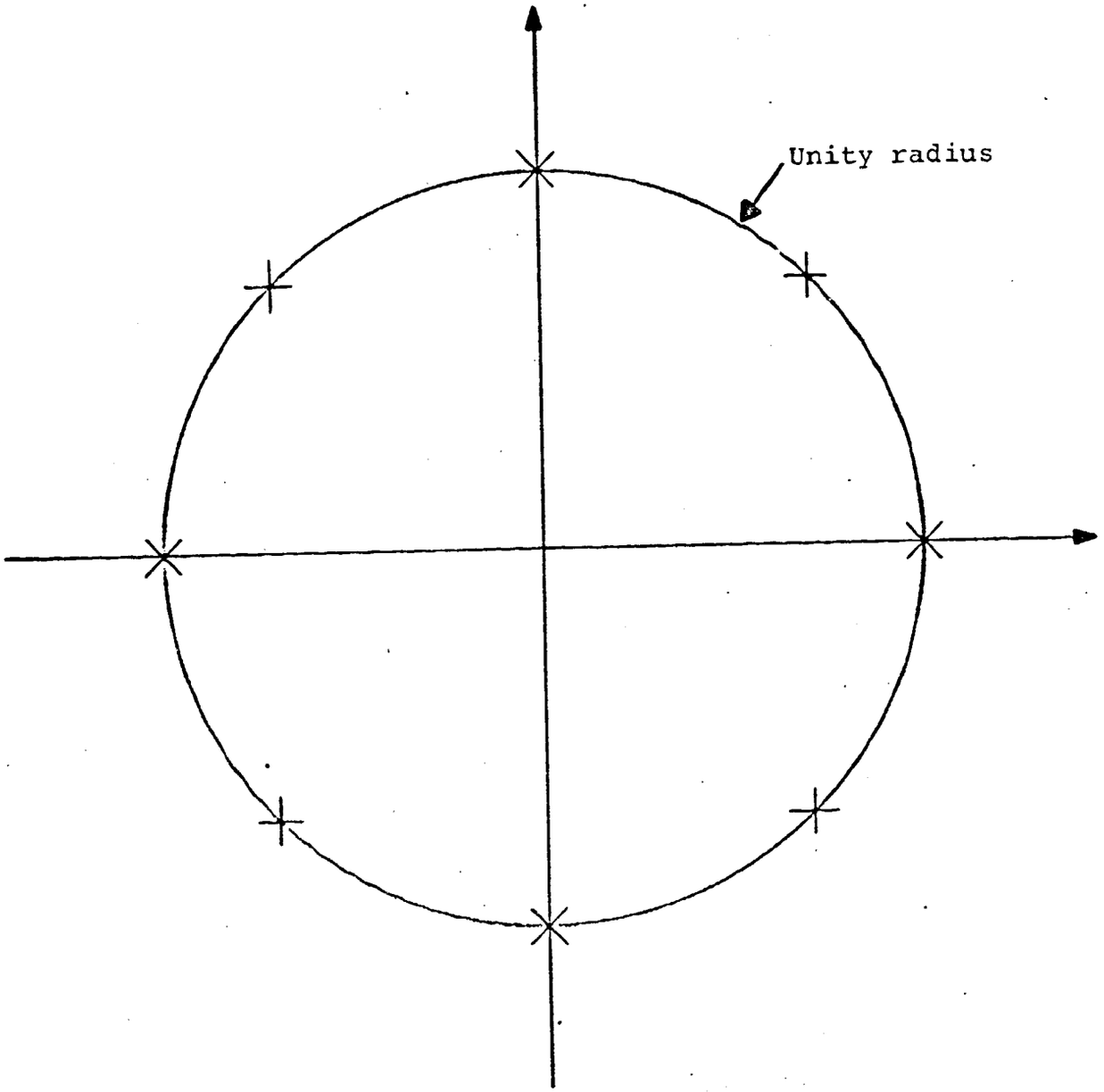


Figure A.1. Normalized Signal Space for the Octal-PSK Modem.

be a vector whose components $N_i^{(m)}$ are equal to the number of times the i^{th} letter, i.e., the signal at 45(i-1) degrees in the signal space, occurs at the m^{th} component in the M code words. Then (A.2) becomes

$$\mu_{\text{total}} = \sum_{m=0}^{n-1} \sum_{i=1}^{M} \sum_{j=1}^{M} \mu\left(C_m(i), C_m(j)\right)$$

$$= \sum_{m=0}^{n-1} \sum_{i=1}^{8} \sum_{j=1}^{8} N_i^{(m)} \, d_{ij} \, N_j^{(m)}$$

$$= \sum_{m=0}^{n-1} N^{(m)} \, D \, N^{(m)^T}$$

$$= M^2 \sum_{m=0}^{n-1} P^{(m)} \, D \, P^{(m)^T} \qquad \text{(A.4)}$$

where $P^{(m)}$ is a probability type vector given by

$$P^{(m)} = \left[\frac{N_1^{(m)}}{M} \quad \frac{N_2^{(m)}}{M} \quad \cdots \quad \frac{N_8^{(m)}}{M} \right]$$

and D is a matrix whose components d_{ij} are the distances squared between the i^{th} and j^{th} letters. Using (A.3)

$$D = \begin{bmatrix} 0 & 2-\sqrt{2} & 2 & 2+\sqrt{2} & 4 & 2+\sqrt{2} & 2 & 2-\sqrt{2} \\ 2-\sqrt{2} & 0 & 2-\sqrt{2} & 2 & 2+\sqrt{2} & 4 & 2+\sqrt{2} & 2 \\ 2 & 2-\sqrt{2} & 0 & 2-\sqrt{2} & 2 & 2+\sqrt{2} & 4 & 2+\sqrt{2} \\ 2+\sqrt{2} & 2 & 2-\sqrt{2} & 0 & 2-\sqrt{2} & 2 & 2+\sqrt{2} & 4 \\ 4 & 2+\sqrt{2} & 2 & 2-\sqrt{2} & 0 & 2-\sqrt{2} & 2 & 2+\sqrt{2} \\ 2+\sqrt{2} & 4 & 2+\sqrt{2} & 2 & 2-\sqrt{2} & 0 & 2-\sqrt{2} & 2 \\ 2 & 2+\sqrt{2} & 4 & 2+\sqrt{2} & 2 & 2-\sqrt{2} & 0 & 2-\sqrt{2} \\ 2-\sqrt{2} & 2 & 2+\sqrt{2} & 4 & 2+\sqrt{2} & 2 & 2-\sqrt{2} & 0 \end{bmatrix} \qquad (A.5)$$

Since the minimum μ for any code must be less than or equal to the average distance squared between non-identical code words, we have

$$\begin{aligned} \mu_{free} \leq \mu_{avg} &= \frac{\mu_{total}}{M^2 - M} \\ &= \frac{M}{M-1} \sum_{m=0}^{n-1} P^{(m)} \, DP^{(m)^T} \\ &\leq \frac{M\,n}{M-1} \, PDP^T \end{aligned} \qquad (A.6)$$

where P is a probability type vector which maximizes the quadratic form $P^{(m)}\ DP^{(m)^T}$ for m=0,1,...,n-1.

The vector

$$P = \left[\frac{1}{8} \quad \frac{1}{8} \quad \cdots \quad \frac{1}{8}\right] \tag{A.7}$$

is at a stationary point of the quadratic form. To show that this is a global maximum we will prove that this quadratic form is a concave function of the vector P.

The eigenvalues of D are $\{\lambda_i\}=\{16,-8,-8,0,0,0,0,0\}$ and, since D is a real symmetric matrix, a set of orthonormal eigenvectors may be chosen with the eigenvector corresponding to the eigenvalue of 16 being

$$x_0 = \left[\frac{1}{\sqrt{8}} \quad \frac{1}{\sqrt{8}} \cdots \frac{1}{\sqrt{8}}\right] \tag{A.8}$$

Using these properties of the D Matrix we will show that the quadratic form is a concave function of the vector P. That is, for any probability type vectors P and Q and positive constants a and b such that a+b = 1

$$(aP+bQ)\ D\ (aP+bQ)^T \geq aPDP^T + bQDQ^T \tag{A.9}$$

To prove (A.9, we expand P and Q in terms of the eigenvectors $\{x_i\}$ of D. With x_0 as the eigenvector corresponding to the eigenvalue of 16, we have

$$P = \sum_{i=0}^{7} P_i x_i = \frac{x_0}{\sqrt{8}} + \sum_{i=1}^{7} P_i x_i \qquad \text{(A.10)}$$

$$Q = \sum_{i=0}^{7} Q_i x_i = \frac{x_0}{\sqrt{8}} + \sum_{i=1}^{7} Q_i x_i \qquad \text{(A.11)}$$

Using (A.10) and (A.11) in (A.9) yields

$$(aP+bQ)\ D\ (aP+bQ)^T = 2 + \sum_{i=1}^{7} \lambda_i\ (aP_i+bQ_i)^2 \qquad \text{(A.12)}$$

$$aPDP^T + bQDQ^T = 2 + a \sum_{i=1}^{7} \lambda_i P_i^2 + b \sum_{i=1}^{7} \lambda_i Q_i^2 \qquad \text{(A.13)}$$

Noting that $\lambda_0 = 16$ is the only positive eigenvalue and using (A.12) and (A.13), (A.9) reduces to

$$\sum_{i=1}^{7} \lambda_i\ (aP_i+bQ_i)^2 \geq \sum_{i=1}^{7} \lambda_i \left(aP_i^2 + bQ_i^2\right)$$

or

$$\sum_{i=1}^{7} |\lambda_i| \left(aP_i^2 + bQ_i^2\right) \geq \sum_{i=1}^{7} |\lambda_i| \left(aP_i+bQ_i\right)^2 \qquad \text{(A.14)}$$

Since x^2 is a convex function of x

$$aP_i^2 + bQ_i^2 \geq \left(aP_i+bQ_i\right)^2 \quad \text{for all i}$$

Equation A.14 is true. Thus the probability vector of (A.7) is a global maximum of the quadradic form PDP^T.

Substituting the maximizing probability vector into (A.6) yields

$$\mu_{free} \leq \frac{2Mn}{M-1} \qquad \text{(A.15)}$$

For this system the number of code words is

$$M = 2^{2L} \qquad , L = 1, 2, \ldots$$

and the bit length of the code words is

$$n = \frac{L}{2} + \frac{K}{2} - 1 \qquad , K = 4, 6, 8, \ldots$$

Minimizing over L for the tighest possible bound yields the desired bound

$$\mu_{free} \leq \min_{L=1,2,\ldots} \frac{2^{2L}}{2^{2L}-1} (L+K-2)$$

REFERENCES

1. J.M. Wozencraft and I.M. Jacobs, Principles of Communication Engineering, Wiley, New York, 1965.

2. C.M. Thomas, "Amplitude-Phase-Keying with M-ary Alphabets: A Technique for Bandwidth Reduction," Proceedings of International Telemetering Conference, Vol. VIII, pp 289-300, October, 1972.

3. K.S. Gilhousen, J.A. Heller, I.M. Jacobs, and A.J. Viterbi, "Coding Systems Study for High Data Rate Telemetry Links," Final Report Prepared for NASA Ames Research Center, Moffett Field, California, under Contract No. NAS2-6024, LINKABIT Corporation, San Diego, California, (Unclassified), January, 1971.

4. D.R. Lumb and A.J. Viterbi, "High Data Rate Coding for the Space Station Telemetry Links," Proceedings of International Telemetry Conference, Vol. VII, pp. 101-106, September 1971.

5. J.A. Heller and I.M. Jacobs, "Viterbi Decoding for Satellite and Space Communication," IEEE Transactions on Communication Technology, Vol. COM-19, pp. 835-848, October 1971.

6. J.A. Heller, "Short Constraint Length Convolutional Codes," Jet Propulsion Laboratory, California Institute of Technology, Space Programs Summary 37-54, Vol. III, pp. 171-177, October/November 1968.

7. J.P. Odenwalder, "Optimal Decoding of Convolutional Codes," Ph.D. dissertation, School of Engineering and Applied Science, University of California, Los Angeles, March, 1970.

8. K.J. Larsen, "Short Convolutional Codes with Maximal Free Distance for Rates 1/3 and 1/4," submitted to IEEE Transactions on Information Theory, June, 1972.

9. LINKABIT Staff, "LINKABIT Corporation Convolutional Encoders-Decoders Applications Information," San Diego, California, 1972.

10. W.F. Schreiber, "Picture Coding," Proceedings of the IEEE, Vol. 55, No. 3, March 1967.

11. A.J. Seyler and Z.L. Budrikis, "Detail Perception after Scene Changes in Television Image Presentations," IEEE Transactions on Information Theory, Vol. IT-11, January 1965.

12. J.B. O'Neal, Jr., "Predictive Quantizing Systems (Differential Pulse Code Modulation) for the Transmission of Television Signals, "Bell System Technical Journal, Vol. 45, May-June 1966.

13. L.G. Roberts, "Picture Coding Using Pseudo-Random Noise," IRE Transactions on Information Theory, Vol. IT-8, February 1962.

14. C.M. Kortman, "Redundancy Reduction - A Practical Method of Data Compression," Proceedings of the IEEE, Vol. 55, No. 3, March 1967.

15. Martin Marietta Corporation, "Study of Potential Applications of Digital Techniques to the Apollo Unified S-Band Communication System," Contract NAS9-9852, NASA Manned Spacecraft Center, Houston, Texas, (Unclassified), 1969.

16. H.J. Landan and D. Slepian, "Some Computer Experiments in Picture Processing for Bandwidth Reduction, "Bell System Technical Journal, Vol. 50, May-June 1971.

17. J.C. Candy, et al, "Transmitting Television as Clusters of Frame-to-Frame Differences, "Bell System Technical Journal, Vol. 50, July-August 1971.

18. P.A. Wintz, "Transform Picture Coding," Proceedings of the IEEE, Vol. 60, No. 7, pp. 809-820, July, 1972.

19. W.K. Pratt, "Spatial Transform Coding of Color Images," IEEE Transactions on Communication Technology, Vol. COM-19, pp. 980-992, December, 1971.

20. J.L. Flanagan, Speech Analysis Synthesis and Perception, Chapter 8, Academic Press, New York, 1965.

21. H.P. Kramer and M.V. Mathews, "A Linear Coding for Transmitting a Set of Correlated Signals," IRE Transactions on Information Theory, Vol, IT-2, pp. 41-46, 1956.

22. C.P. Smith, "Voice-Communications Method Using Pattern Matching for Data Compression," J. Acoust. Soc. Am., Vol. 35, p. 805, 1963.

23. J.P.M. Schalkwijk, "An Algorithm for Source Coding," IEEE Transactions on Information Theory, Vol IT-18, pp 395-399, May 1972.

24. J.M. Cover, "Enumerative Source Encoding", to be published IEEE Transactions on Information Theory, Vol. IT-19, January 1973.

25. E.R. Berlekamp, Algebraic Coding Theory, McGraw-Hill, New York, 1968.

26. I.M. Jacobs and J.A. Heller, "Performance Study of Viterbi Decoding as Related to Space Communications," Final Technical Report, Contract No. DAAB07-71-C-0148, U. S. Army Satellite Communications Agency, (Unclassified), August, 1971.

27. I.M. Jacobs and R.J. Sims, "Configuring a TDMA Satellite Communication System with Coding," Conference Proceedings of the Fifth Hawaii International Conference on System Sciences, Honolulu, Hawaii, January, 1971.

28. I.L. Lebow, K.L. Jordan, Jr., and P.R. Drouihet, Jr., "Satellite Communications to Mobile Platforms," Proceedings IEEE, Vol. 59, pp. 139-159, February 1971.

29. J.G. Puente, W.G. Schmidt, and A.M. Werth, "Multiple Access Techniques for Commercial Satellites," Proceedings IEEE, Vol. 59, pp. 218-229, February 1971.

30. A.R. Cohen, J.A. Heller, and A.J. Viterbi, "A New Coding Technique for Asynchronous Multiple Access Communications," IEEE Transactions on Communication Technology, Vol. 19, pp. 835-848, October 1971.

31. J.P. Odenwalder, K.S. Gilhousen, J.A. Heller, J.M. Jacobs F. Jelinek, A.J. Viterbi, "Hybrid Coding Systems Study," Final Report Prepared for NASA Ames Research Center, Moffett Field, California, under Contract No. NAS2-6722, LINKABIT Corporation, San Diego, California, (Unclassfied),, September 1972.

END

DATE

FILMED

JUN 5 1973