SECOND AND THIRD
INTERIM REPORT

February 15, 1970 – June 1970

DIGITAL COMMUNICATIONS TECHNIQUES

for

National Aeronautics and Space Administration

under

NASA GRANT NGR-33-006-020

Prepared by

Raymond L. Pickholtz, Associate Professor

PIBEE 70-[redacted]/003

1970

DEPARTMENT OF ELECTRICAL ENGINEERING
POLYTECHNIC INSTITUTE OF BROOKLYN

A Coding Technique for a Dispersive Channel

I. INTRODUCTION

The problem considered here is that of reducing intersymbol interference caused by time dispersive channels and thereby also reducing the error probabilities associated with transmitting digital data over such channels. This problem was considered by Nyquist (Ref. 1) as early as 1928, and much recent work on this subject has centered on the use of the Tapped Delay Line (TDL) Equalizer (Refs. 2 to 7). The conventional TDL equalizer approach is to adjust the tap gains to minimize a measure of the error between channel input and equalizer output. Typically the mean squared error is minimized. In essence this approach chooses the non-dispersive channel response as the desired overall response of channel plus equalizer.

The approach described here recognizes the encoding properties of time dispersive channels. These channels process the transmitted data in much the same way as the generator of a cyclic algebraic code. The coefficients of a code generator, which closely match the channel path gains, are chosen as the desired channel for adjusting an otherwise conventional TDL equalizer. This has been termed coded equalization.

Trnasmission of k q-ary symbols through the dispersive channel in cascade with the coded equalizer results in an approximation to a code word. Error correction can thus be obtained using an ordinary decoder and without transmission of parity symbols.

Error bounds and results of computer simulations show a potential for significant reductions in error rate using relatively short TDL equalizers.

II. CODED EQUALIZATION

A discrete time version of the time dispersive channel is shown in

Figure 1. The channel sample response values a_0, a_1, --- a_{M-1} may be thought of a s path gains. The time spread is MT seconds where T is the duration of an input symbol. We assume that after MT seconds the channel response can be ignored.

Use of a minimum mean squared error TDL equalizer yields a residual distortion power given by

$$\sigma_Z^2 = \sigma_\alpha^2 \sum_i (h_i - d_i)^2 \tag{1}$$

where σ_α^2 is the signal variance, h_i is the overall response of the channel plus equalizer and d_i is the desired overall response. For a conventional equalizer d_i is chosen as the Kronecker delta and the desired response is that of a nondispersive channel. It has been shown (Ref. 7) that the optimum tap gains are given by

$$\bar{q} = (A^T \ A)^{-1} \ A^T \ \bar{d} \tag{2}$$

where A is the Channel matrix given by

$$A = \begin{bmatrix} a_0 & & & \\ a_1 & a_0 & & \\ \vdots & a_1 & \ddots & a_0 \\ a_{M-1} & & \ddots & a_1 \\ & a_{M-1} & & \vdots \\ & & \ddots & \vdots \\ & & & a_{M-1} \end{bmatrix}_{J \times N} \tag{3}$$

$\bar{d}$ is the desired response in vector form, N is the number of equalizer taps, and $J = N + M - 1$.

Note that Eq. 2 gives the optimum tap gains for any <u>given</u> desired response. The conventional choice of d_i as the Kronecker delta has the convenience of yielding the data directly, but may also yield a large value of σ_Z^2. Clearly a choice of d_i closer to a_i will decrease σ_Z^2, but will result in an encoding of the data. The problem is to find such a desired response which allows for efficient decoding.

To find a possible choice for d_i, we consider certain properties of cyclic block codes. The encoding operation can be represented in matrix form by

$$\overline{C} = \overline{\alpha}\ G \tag{4}$$

where $\overline{\alpha}$ is the k-symbol data vector, $\overline{C}$ is the n-symbol output code vector, and G is an n x k generator matrix given by

$$G = \begin{bmatrix} g_0 & & \\ g_1 & g_0 & \\ \vdots & g_1 & \ddots \\ g_r & \vdots & \ddots & g_0 \\ & g_r & \ddots & g_1 \\ & & \ddots & \vdots \\ & & & g_r \end{bmatrix} \tag{5}$$

The matrix elements g_i are the coefficients of the code generator polynomial (Ref. 8).

We now wish to compare the time dispersive channel with this code generator. Suppose the channel dispersion is $M = r + 1$, and that we transmit k symbols. The channel would then output n symbols where $n = k + r$. In vector form the output is

$$\overline{y} = \overline{\alpha}\ A' \tag{6}$$

where A' is the same as A except its dimensions are n x k. The similarity between the time dispersive channel and a cyclic encoder is now obvious. The main difference is that Eq. 4 implies operations in a finite field with q elements whereas the operations of Eq. 6 require ordinary arithmetic. However, by constraining the transmission alphabet size q, to be a prime number, this apparent difficulty is overcome. Thus if $a_i = g_i$ for $0 \le i \le r$, then reduction modulo q at the receiver would yield the same output (in the absence of noise) as would the cyclic encoder.

Of course in general $a_i \neq g_i$ and the channel will not match the code perfectly. But if a code is found to give a close match, the channel output will be very similar to a code word. A TDL equalizer can then be used with the tap gains given by

$$\bar{q} = (A^T A)^{-1} A^T \bar{g} \tag{7}$$

The residual distortion becomes

$$\sigma_Z^2 = \sum_{i=1}^{J} (h_i - g_i)^2 \tag{8}$$

The data transmission is intermittent as is the case when using an error correcting code. However, in this case we simply interrupt the transmitter for an r-symbol period instead of transmitting the parity symbols.

III. ERROR BOUND

An upper bound was derived using the Chernoff bounding procedure and based on the work of Lugannani (Ref. 9) but with several important differences. Lugannani assumed a binary, independent and stationary input sequence. In the present case, due to the intermittent format, the input sequence is neither stationary or independent. In addition, we assume q-ary input symbols (q equal to a prime) and impose the parameters n and k of the code. The bound on probability of error applies to the decoder input. Thus the error correction capability of the code provides an additional improvement. The error bound which is derived in Ref. 10, is given here in terms of tabulated error functions,

$$Pe \leq K \left[\tfrac{1}{2}(1- \operatorname{erf} (E/2-\lambda)/\sigma_o) + \tfrac{1}{2} (1- \operatorname{erf} (E/2 + \lambda)/ \sigma_o)\right]$$

$$+ 2K\, B \exp\left[-E^2(1-B^2)/8\, \sigma_o^2 \right] \{ \tfrac{1}{2}(1- \operatorname{erf}(\lambda /B -EB/2)/\sigma_o)$$

$$-\tfrac{1}{2}(\ 1\text{-erf}\ (D/B\text{-}EB/2)/\sigma_o\ )\ -\tfrac{1}{2}(\ 1\text{-erf}\ (\lambda\ /B + EB/2)/\sigma_o)$$

$$+\tfrac{1}{2}\ (1\text{- erf}(D/B + EB/2)/\sigma_o)\} \qquad (9)$$

where E is the distance between adjacent output levels/ σ_o^2 is the noise variance at the equalizer output, D is the peak distortion and K is a function of the a priori probabilities of reaching the extreme output levels. The parameters B^2 and λ are given by

$$B = \frac{\sigma_Z^2}{\sigma_Z^2 + (ak/2nq)\ \sigma_o^2} \qquad (10)$$

$$\lambda = \sigma_Z\ [\ 2nq\ \ \ln 4/ak\ ]^{\frac{1}{2}} \qquad (11)$$

In order to see the effect of intersymbol interference above, we evaluate Eq. 9 in the limit of decreasing σ_Z^2.

$$\lim_{\sigma_o^2 \to o} Pe \leq \begin{cases} K, & E/2 < \lambda < D \\ 2K \exp\ (-E^2\ ak/\ 16nq\ \ \ \sigma_Z^2), & \lambda < E/2 < D \\ 0, & \lambda < D < E/2 \end{cases} \qquad (12)$$

We see that the error probability will go to zero if the peak distortion D is less than the threshold value, E/2. Also if $\lambda < E/2 < D$ the error probability bottoms at a value dependent on the code parameters and the distortion variance. This type of performance is also obtained with the conventional equalizer. In fact, if we let q = 2 (binary transmission) and n = k (no code) then Eq. 9 is the same as derived in Res. 9. The significant factors here are the values of D and σ_Z^2. If the channel path gains are closely matched by the code coefficients, these values will be much reduced. Thus the coded equalizer offers a potential for reducing error probabilities beyond that achieved with conventional equalizers. It should be noted the factor E^2 ak/nq increases with decreasing code rate k/n.

IV. SIMULATED PERFORMANCE

In order to establish examples of system performance and to confirm the derived error bound, computer simulations of both the conventional and coded equalizer systems were performed. The assumed channel response shown in Figures 2 and 3 are typical of telephone cables and referred to as channel "a" and channel "b" respectively. For the purpose of the simulations these channels were considered time invariant. This is justified since known tracking algorithms (Refs. 5 and 7) apply to the assumed equalizers. The channel sample values are as follows:

$a = (.15, -.78, 1, -.7, -1, .05, .95, .07, .12, .05)$

$b = (.07, -.15, 1, -.09, .78, -1, .125, -.67, .1, 05)$

where the channels are normalized such that the strongest path gain is unity. The equalizer tap gains were calculated from Eq. 2 for the conventional system and from Eq. 7 for the coded system. The code chosen for channel "a" is a (8, 3) BCH code with generator coefficients

$$(g_0, g_1, g_2, g_3, g_4, g_5) = (-1, 1, -1, -1, 0, -1)$$

For channel "b" a (12, 7) fire code is chosen with generator coefficients

$$(g_0, g_1, g_2, g_3, g_4, g_5) = (1, 0, 1, -1, 0, -1)$$

In both cases q = 3 and the coefficient 2 has been replaced by its modulo 3 equivalent, -1. In the conventional case α_j the values ± 1 with equal probability. In the coded case α_j takes on the values $0, \pm 1$ equi-probably during the k symbol interval of data transmission. During the r symbol gap $\alpha_j = 0$. For all transmissions appropriate scaling is used so that the transmitted signal variance is unity.

For both systems, 10 tap and 7 tap equalizers were used on channels "a" and "b" respectively. Results are plotted as a function of the signal-to-additive noise ratio at the receiver input to facilitate comparison between the two systems.

Figures 4 and 5 show the simulation results and the derived upper bound for the coded equalization system. The bound is on the symbol error rate before decoding. In Figures 6 and 7 the results for the conventional and coded systems are plotted together for easier comparison. For low signal-to-additive noise ratios, the conventional system shows better performance. However, in this noise dominated region the coding improvement is small or even negative. At higher signal-to-noise ratios the error rate can be significantly reduced by the coded system. For example, in Figure 7 the coded system shows 3 orders of magnitude improvement at 15 db using a 7 tap equalizer. At this point the conventional system has already bottomed, whereas the error rate for the coded system approaches zero in the limit of decreasing additive noise. In Figure 6 the simulation curve was theoretically extended beyond the actual simulation. This was done because excessive computer time is required to measure error rates much below 10^{-4}. The theoretic extension was calculated by assuming the measured symbol errors (before decoding) occurred randomly. The word error rates indicated for the coded system are equaivalent to assuming that when a word is in error, every symbol in the word is in error. This is obviously not the case, and so the actual symbol error rate is below the plotted word error rate.

The results given here show that, for a given error rate, it may be possible to use a much shorted tapped delay line than for the conventional system.

An important consideration is the effect of mismatch between the channel and code. To get an indication of this effect the (12, 7) Fire Code was simulated on channel "a". Figure 8 shows sample response functions of the channel, the code, the equalizer tap gains, and the final overall output response. The mismatch is quite obvious and yet the measured loss in signal to distortion ratio was about 3 db. Referring to Eq. 7 it is clear that if the channel and code matched perfectly, the equalizer tap gains would all be zero except for one which would be unity. Thus the distribution of tap gains gives a measure of channel code mismatch.

V. CONCLUSIONS

We have treated the time dispersive channel as a linear encoder. In this approach an algebraic cyclic code generator is matched to the channel response. It was demonstrated by means of computer simulation, using typical telephone line response functions, that orders of magnitude improvement in error probability may be achieved. In particular, in certain cases where the conventional equalizer system error rate bottoms, the coded equalizer system error rate goes to zero with decreasing additive noise variance.

The upper bound on the probability of error developed by Lugannani (Ref. 9) has been extended to inclue q-ary, non-stationary, dependent message sequences. The particular form of dependence is introduced by a message format such as used in error control encoding.

REFERENCES

1. Nyquist, H., "Certain Topics in Telegraph Transmission Theory", Trans. AIEEE, Vol. 47, 617-644, April 1928.

2. George, D. A. and Coll, D. C., "The Recption of Time Dispersed Pulse", 1st IEEE Annual Communication Convention Record, 749-752, June 1965.

3. Ditoro, M. J., "A New Method of High-Speed Adaptive Serial Communication Through Any Time Variable and Dispersive Transmission Medium", 1st Annual Communication Convention Record, 763-767, June 1965.

4. Aaron, M. R. and Tufts, D. W., "Intersymbol Interference and error Probability", IEEE Transaction, IT-12, 26-34, January 1966.

5. Niessen, C. W. and Drouilhet, P. R., Jr., "Adaptive Equalizer for Pulse Transmission", Lincoln Laboratory, MIT, May 1967.

6. Lucky, R. W. and Rudin, H. R., "An Automatic Equalizer for General Purpose Communication Channels", BSTJ, Vol. 46, 2179-2208, November 1967.

7. Gersho, A., "Adaptive Equalization of Highly Dispersive Channels for Data Transmission: I" BSTJ, Vol. 48, January 1969.

8. Peterson, W. W., "Error-Correction Codes", The MIT Press and John Wiley & Sons, Incorporated, 1961.

9. Lugannani, R., "Intersymbol Interference and Probability of Error in Digital System", IEEE Trans., IT-15, 682-688, November 1969.

10. Klein, T. J. "The Time Dispersive Channel as a Linear Encoder", Dissertation, P. I. B., June 1970.

B Applications of Recursive Techniques

Adaptive array processing

The problem is to automatically make an array of isotropic detectors form a beam in a desired direction in space when unknown interfering noise is present so as to maximize the signal-to-noise ratio (SNR). Iterative gradient techniques are used to do this.

One question that immediately arises is what approach do we use, i.e. we can view the detectors either as an antenna array, and, by using the concept of an antenna pattern, solve for those current and phase excitations which produce the maximum SNR, or, alternatively, we can view the detector arrary as a multichannel filter, and, using statistical communication theory, solve for those filter coefficients which maximize the SNR. However, since there is only one physical problem, these two different approaches must ultimately yield equivalent results. This was demonstrated, and moreover, when the noise is monochromatic, it turned out that an equivalence could be shown to exist even between intermediate terms in the two approaches. Thus, this phase of the research was concerned with demonstrating the equivalence between the "antenna pattern" and "multichannel filter" points of view in designing optimum arrays. Specifically we investigated:

1. Optimum array design using the "antenna pattern" point of view, assuming the incident noise power is known.
2. Optimum array design using the "multichannel filter" point of view, assuming the noise space-time correlation function is known.
3. The general relationship between the space-time correlation function and the incident noise power.
4. The equivalence of parts 1 and 2 above under a monochromatic noise assumption.

Next, using the "antenna pattern" point of view we investigated the sensitivity of the SNR to random excitation errors and random errors in the detector locations. This sensitivity is essentially given by the super-gain ratio, and through the use of the analogy described above, we were able to find an analog to the super-gain ratio in terms of communication theory quantities (e. g. correlation functions). It was noted that when we use linear arrays of detectors separated by one half wavelength or less, this sensitivity factor became very large when the optimum currents of phases of part 1 above were used, thus indicating that we should not try to design our antenna pattern or multichannel filter coefficients on the basis of maximizing the SNR alone, but rather on the basis of maximizing the SNR subject to a constraint on the supergain ratio as done by Lo, Lee, and Lee (Proceedings of IEEE, vol 54, no 8, August 1966). Numerical determination of the optimum excitations to use when we constrain the super-gain ratio is now being investigated.

Next, we tried to analytically consider adaptive algorithms which would maximize the SNR subject to a constraint on the super-gain ratio when unknown interfering noise is present. Because the SNR and super-gain ratio are nonlinear quantities, it turned out to be exceedingly difficult to prove convergence of the algorithms' rates of convergence. Thus, solely for the purpose of mathematical tractability (the nonlinear problem will be simulated on a computer to obtain some numerical intrication of convergence rates), we considered adaptive algorithms which minimize the mean square error (MSE) subject to a linear constraint. Specifically, we found the Lagrange solution to the problem of minimizing the MSE subject to a linear constraint and then prove that an algorithm of the form converges to the Lagrange solution in real time, with an easily expressible bound on the convergence rate, where k is the step size constant, P is a matrix projection operator (see J. B. Rosen, Journal of SIAM, vol 8 March 1960 and vol 9 December 1961), and $D_{\underline{w}j}$ is the gradient of the MSE with respect to $\underline{W}_j$. We proved the convergence and found bounds on the rate of convergence when the gradient was (1) known exactly (2) estimated, and (3) estimated by a noisy estimate.

We next plan to simulate these algorithms on a computer and compare the similated and theoretical results.

C A Decoding Algorithm for Truncated Convolutional Codes

I. INTRODUCTION

The computational effort exerted by an ideal maximum likelihood decoder is fixed and grows exponentially with the code block length. By contrast the number of computations required by a sequential decoder is a random variable depending on the channel noise level.[1,2]

On the mean, the sequential decoder will exert a significantly low computational effort as long as the source rate is lower than a certain rate called R comp. The decoding error using sequential decoding methods approximates the error of the maximum likelihood decoder, i. e. it decreases exponentially with the constraint length of the code.

Several decoding algorithms were suggested. The first algorithm was due to Wozencraft[(3)], and offered a computational effort which grows as a small power of the constraint length. This algorithm was followed by Fano's[(4)] decoding scheme which is easy to implement and exhibits a mean computational effort which is independent of the constraint length.

A different algorithm was recently introduced by Zingagirov and Jelinek[(5)]. Unlike Fano's algorithm this decoder stores in memory the address and the likelihood function value (l. j. v) of all the nodes that were visited during the decoding process but were not extended. This data storage eliminates the need for a search whenever a wrong path is suspected. As a result the computational effort is decreased at the expense of increasing considerably the decoder's memory.

Another interesting algorithm was proposed by Viterbi. This scheme which was proven to be a maximum likelihood scheme, offers a reduction, although not a significant one, in the computational effort, and also offers a more efficient bound on the probability of error by taking advantage of the characteristic inherent to tree codes.

The algorithm proposed in this report is of a tree searching type, and in gross features resembles the Fano algorithm. The differences are clearly evident in the initiation and execution of the search procedure when a wrong path is suspected.

To sum up the main differences:

(1) In the new scheme a search will be initiated only when a decrease in the l. p. v. of the path followed by the decoder exceeds a chosen value ϵ, while using Fano's algorithm a search will be initiated when the decrease in the l. p. v. exceeds a random value Δ confined to the interval $0 \leq \Delta \leq \epsilon$.

(2) Upon initiating a search at a node at depth j, the proposed scheme will end up finding an accessible node at depth j having the largest l. p. v. and it will resume to the normal decoding procedure from that node. Therefore, only a path having the highest probability of being correct will be extended.

II. TREE ENCODING

The received data will be fed into the decoder in blocks of N channel digits. The encoder constructs a block code from a treee code by terminating it as follows:

The encoder accepts a block of K successive source information letters. The K input letters are transformed into an output sequence which forms a path in a K-level code tree pertinent to the encoder used. Each block of K information digits is succeeded by t additional predetermined digits which form a terminating sequence. The code tree is illustrated in Fig. 2. After the insertion of the t terminating sequence input digits, the encoder is reset and is realy for a new block of K information digits.

Using a code which posses a q-Ary alphabet with b channel input digits per branch, our block length will be

$$N = b(K + t)$$

with $M = q^K$ possible massages the block code rate will be:

$$R = \frac{\ell n M}{N} = \frac{K}{K+t} \quad \frac{\ell n q}{b} = \frac{K}{K+t} \quad r$$

where r is the tree code rate.

The net transmission rate will be reduced by $\frac{K}{K+t}$. We can minimize the effect of decreasing rate by choosing $K >> t$.

It will be shown later that the probability of decoding an error will be a decreasing exponential function of t.

III. THE DECODING ALGORITHM

The measure used throughout this report for comparing the paths in the code tree will be the likelihood function as defined by Fano.

The decoding steps will be the following:

1) Starting at the root node, compute the branch likelihoods of the q branches leaving that node. The branch possessing the largest branch likelihood will be tentatively chosen as the first branch of the correct path.

2) Assign a threshold T_1:

$$T_1 = L_{m(1)}$$

where $L_{m(1)}$ denoted the l.f. of the mth path ending at a node at depth i.

3) Proceed by computing the branch likelihoods of the q branches leaving the node presently occupied by the decoder and choose the branch yielding the largest branch likelihood. The likelihood function of the extended path will now be $L_{m(2)}$.

4) Make the following comparisons:

a) $L_{m(2)} - T_{1} \gtrless 0$

b) $L_{m(2)} - T_{1} \gtrless -\epsilon$

where ϵ is arbitrarily chosen.

5) a) If $\begin{cases} L_{(2)} - T_1 > -\epsilon \\ L_{(2)} - T_1 > 0 \end{cases}$

increase threshold to $T_2 = L_{m(2)}$ and proceed to decode a branch at depth 3.

b) If $\begin{cases} L_{m(2)} > T_1 > -\epsilon \\ L_{m(2)} > T_1 < 0 \end{cases}$

leave threshold $T_2 = L_{m(1)}$ unchanged and proceed to decode a branch at depth 3.

This same procedure which will be called the "regular" procedure will be applied to the succeeding nodes 3, 4, . . . j. The "regular" procedure will be interupted at node j when:

c) $L_{m(j)} - T_{j-1} < 0$

$L_{m(j)} - T_{j-1} < -\epsilon$

In this case a search procedure is initiated.

6) Start the search procedure by taking the following steps:

a) Declare an intermediate threshold $T_j^1 = L_{m(j)}$ and store in memory the address and threshold T_j belonging to the node $S_{m(j)}$ which initiated the search procedure.

b) Visit all accessible nodes at a depth smaller or equal to j. An accessible node will be a node $S_{n(x)}$ $0 \leq \alpha \leq j$ that can be reached from node $S_{m(j)}$ and does not violate the threshold T_j^1.

c) If no node at depth j is penetrated, proceed with the "regular" procedure from the node which initiated the search.

d) If other nodes at depth j were penetrated in the course of the "search" procedure, proceed with the "regular" procedure from the node at depth j which exhibits the largest likelihood function value.

7) Extend the path into the code tree by using the "regular" and "search" procedures alternatively as required.

8) Terminate the decoding procedure when the decoder reacher the last node $\Gamma = K + t$ by means of the "regular" procedure and:

$$L_{m(\Gamma)} - L_{m(\Gamma > 1)} > - \epsilon$$

or when the decoder reaches node Γ by means of a "search" procedure.

The flow chart for the algorithm is given in Fig. 3.

IV. UNDETECTABLE PROBABILITY OF ERROR

The upper bound on the probability of error is derived by using the following lemma:

Lemma 1

A necessary but insufficient condition for an error to occur at a node at depth i is:

$$L^{\alpha}_{m(\Gamma)} - \epsilon \geq \min_{i+1 \leq j \leq \Gamma} L_j$$

Using this lemma we can express the probability of error at a mode of depth i

$$P_e(i) \leq P_r \{ \bigcap_{m \in D_i} [L^{\alpha}_{(\Gamma)} - \epsilon \geq \min_{i+1 \leq j \leq \Gamma} L_j] \}$$

which leads to a mean probability of error

$$\overline{P}_e \leq C\, e^{-b+R\,comp}$$

where: $C = [\, A^2(e) + B^2\,(R)]\, e^{\epsilon/2}$

$$A(R) = \sum_{m=1}^{\infty} e^{\frac{-bm}{2}[R comp - R]}$$

$$B(R) = \sum_{j=1}^{\infty} e^{-\alpha b\,[R comp - R]}$$

V. COMPUTATIONAL EFFORT

The upper bound on the mean number of decoding steps in the incorrect subset emanating from node i is found by using the two following lemmas.

Lemma 2:

The threshold of the decoder will never be lowered below

$$T > \begin{cases} \min\limits_{i+1<j\leq\Gamma} L_j - A_o & A_o > \epsilon \\ \min\limits_{i+1<j\leq\Gamma} L_j - (A_o + \epsilon) & A_o < \epsilon \end{cases}$$

Lemma 3

The node $S_{m(e)}$ will be visited by the decoder for the ith time if:

$$L^{\alpha}_{m(e)} \geq \min_{e+1\leq j\leq\Gamma} L_j + (i-1)\;\epsilon - A_o$$

Using the above mentioned lemmas we can express the mean number of computations on all nodes in the incorrect subset stemming from node j as:

$$N_j \leq B\;\; [\sum_{e=o}^{\Gamma=j} \exp - b^e\, [R\ comp - R]\,]^2$$

where:
$$B = \begin{cases} \dfrac{e^{A_o/2}}{1-e^{-\epsilon/2}} & A_o = \epsilon \\ \dfrac{e^{\epsilon/2}}{1-e^{-\epsilon/2}}\, e^{A_o/2} & A_o < \epsilon \end{cases}$$

VI. SIMULATION

A computer simulation will be carried out in order to evaluate the decoding effort of the algorithm.

VII <u>NOTATION</u>

A_o - Maximum value attained by a branch likelihood function

b - Number of digits in a branch

k - Number of information digits in a block

$L^{\alpha}_{m(j)}$ - Likelihood function of the m^{th} path at depth j in the incorrect subset

L_j - Likelihood function of the correct path at depth j

N - Block length

R - Block code rate

r - Tree code rate

T_j - Threshold at node at depth j

t - Number of digits in terminating sequence

REFERENCES

1. J. M. Wozencraft and L. M. Jacobs - "Principles of Communication Engineering", John Wiley 1965.
Engineering", John Wiley 1965.
2. R. G. Gallager - "Information Theory and Reliable Communication" John Wiley 1965.

3. J. M. Wozencraft and B. Reiffen - "Sequential Decoding", M. I. T. Press, Cambridge, Mass. 1961.

4. R. M. Fano - "A Heuristic Discussion of Probabilistic Decoding" IEEE, Trans. IT-9, 64-73, 1963.

5. F. Jelinek, - "A Stack Algorithm For Faster Sequential Decooling" Report Rc 2441, IBM Research Division, Yorktown Heights, N. Y. 1969.

6. A. J. Viterbi - "Error Bounds on Convolutional Codes, and An Assymptotically Optimum Decoding Algorithm", Trans. IT-13 April 1967.

D Equalization of Partial Response Channels

The preset equalizer and an adaptive equalizer have been determined for the duobinary data transmission scheme.

The duobinary technique utilizès intersymbol interference in a constructive manner. Assuming an ideal channel, the duobinary input signal to the receiver is

$$y_k = a_k + a_{k-1}$$

where the symbols, a_k, are the precoded data. The $a_{k's}$ are chosen from an m-level alphabet; the receiver is a modulo m detector. For a more realistic channel, the channel characteristic will distort the incoming data pulses producing unwanted intersymbol interference. The input to the receiver then is

$$y_k = a_k x_0 + a_{k-1} x_1 + \sum_{n \neq k, k_{-1}} a_k x_{k-n}$$

where the last term represents the additional interference.

The transversal filter equalizer (input Z_n, output $x_n = C^+ Z_n$) is used to eliminate the unwanted intersymbol interference. This is done by adjusting the filter coefficients, C_i, to minimize the mean square error subject to constraints on x_0 and x_1. Using Lagrange multipliers the coefficient vector, C, is given by

$$C = \frac{1}{2} A^{-1} (\alpha_0 Z_0 + \alpha_1 Z_1) \tag{1}$$

with

$$A = \sum_n Z_n Z_n^+$$

$$\alpha_0 = 2(Z_1 - Z_0, A^{-1} Z_1)/[(Z_1, A^{-1} Z_1)(Z_0, A^{-1} Z_0) - (Z_0, A^{-1} A_1)^2]$$

$$\alpha_1 = 2(Z_0 - Z_1, A^{-1} Z_1)/[(Z_1, A^{-1} Z_1)(Z_0, A^{-1} Z_0) - (Z_0, A^{-1} Z_1)^2]$$

Since the channel is usually not completely known, an adaptive version of the equalizer is necessary. The coefficient solution is not applicable for use during a training period because the constants α_0 and α_1 cannot be determined without prior knowledge of the channel. To circumvent this difficulty, penalty functions are used to augment the mean square error. The penalty functions will increase the performance measure for violations of the constraints. In our problem the penalized performance measure is

$$P_p = \sum_n \epsilon^2_n + N_0\epsilon^2_0 + N_1\epsilon^2_1$$

where

$$\epsilon_n = \begin{cases} x_n & n \neq 0,\ 1 \\ x_{n-1} & n = 0,\ 1 \end{cases}$$

with N_0 and N_1 positive numbers.

For each permissible value of N_0 and N_1, a unique minimum exists; the corresponding filter coefficient vector is a solution of

$$[A + N_0 Z_0 Z_0^+ + N_1 Z_1 Z_1^+]C = (N_0 + 1)\ Z_0 + (N_1 + 1)Z_1 \qquad (2)$$

As N_0 and N_1 approach infinity, the vector solution of (2) reduces to the Langrange solution. For finite weights, the penalty functions will force the solution to be in a neighborhood about the constraints. Furthermore, to be in a specified neighborhood, there exist semi-infinite intervals for the values of N_0 and N_1. With N_0 and N_1 finite, Eq. (2) lends itself to an iterative method of solution. Using the gradient algorithm we adjust the coefficients in manner such as to drive the gradient of the penalized measure to zero.

$$C^{(k+1)} = (I - \underline{\underline{\alpha}}\ B)\ C^k + \alpha(N_0 + 1)\ Z_0 - \alpha\ (N_1 + 1)Z_1$$

with

$$B = A + N_0 Z_0 Z_0^+ + N_1 Z_1 Z_1^+$$

Convergence of the algorithm is guaranteed for $0 < \alpha < 2/\lambda_{max}$. Simulations have indicated that this is indeed true.

Future research will be directed into increasing the rate of convergence of the above scheme and investigating a quadratic programming approach. Also the equalizer problem will be generalized to the class of partial response channels.

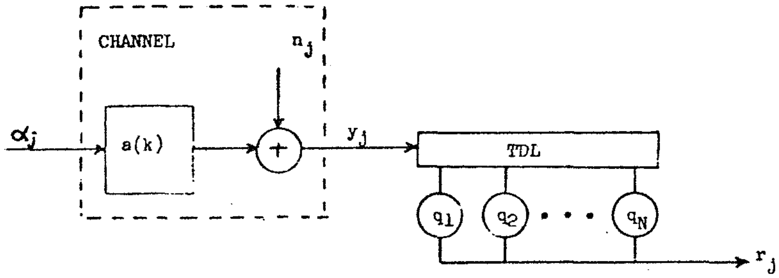
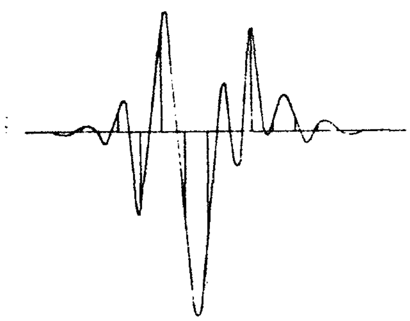


Fig. 1 Channel and Equalizer



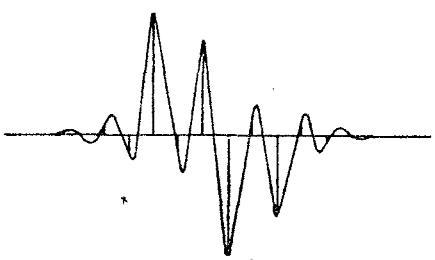


Fig. 3 Channel "b"

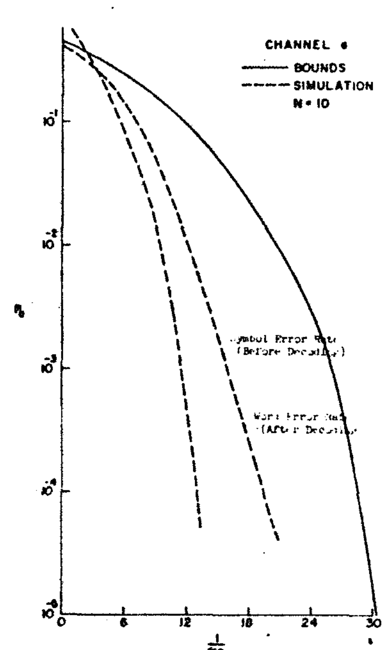


FIG 4 CODED EQUALIZER SYSTEM

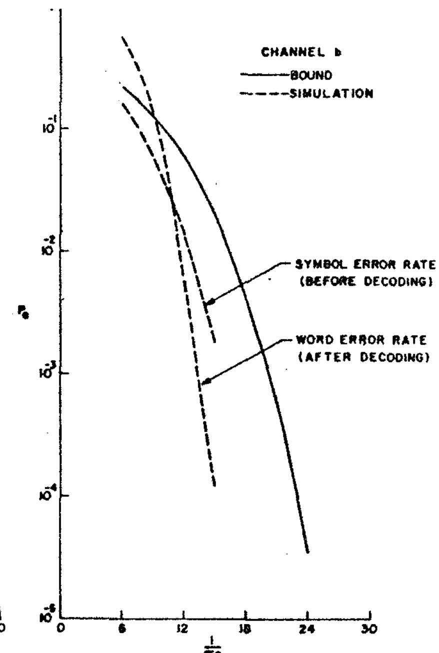


FIG. 5 CODED EQUALIZER SYSTEM

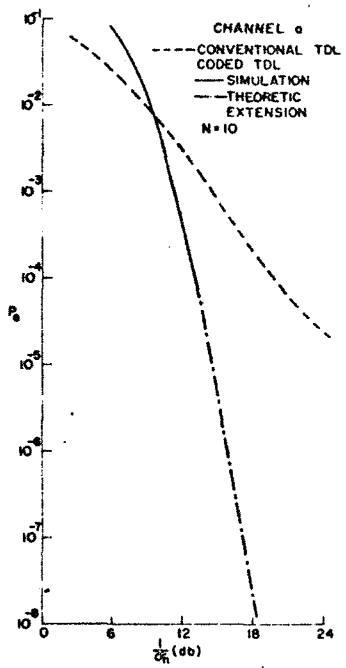


FIG. COMPARISON OF CONVENTIONAL AND CODED SYSTEM PERFORMANCE

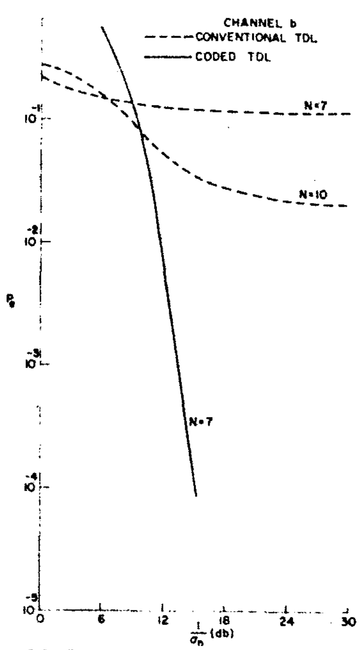


FIG 7 COMPARISON OF CONVENTIONAL AND CODED SYSTEM PERFORMANCE

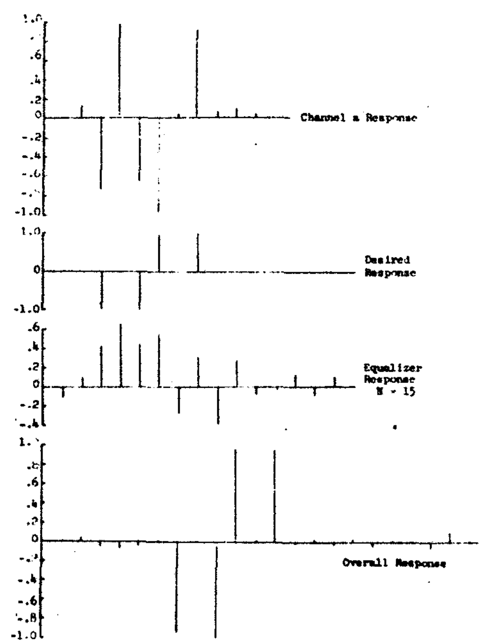
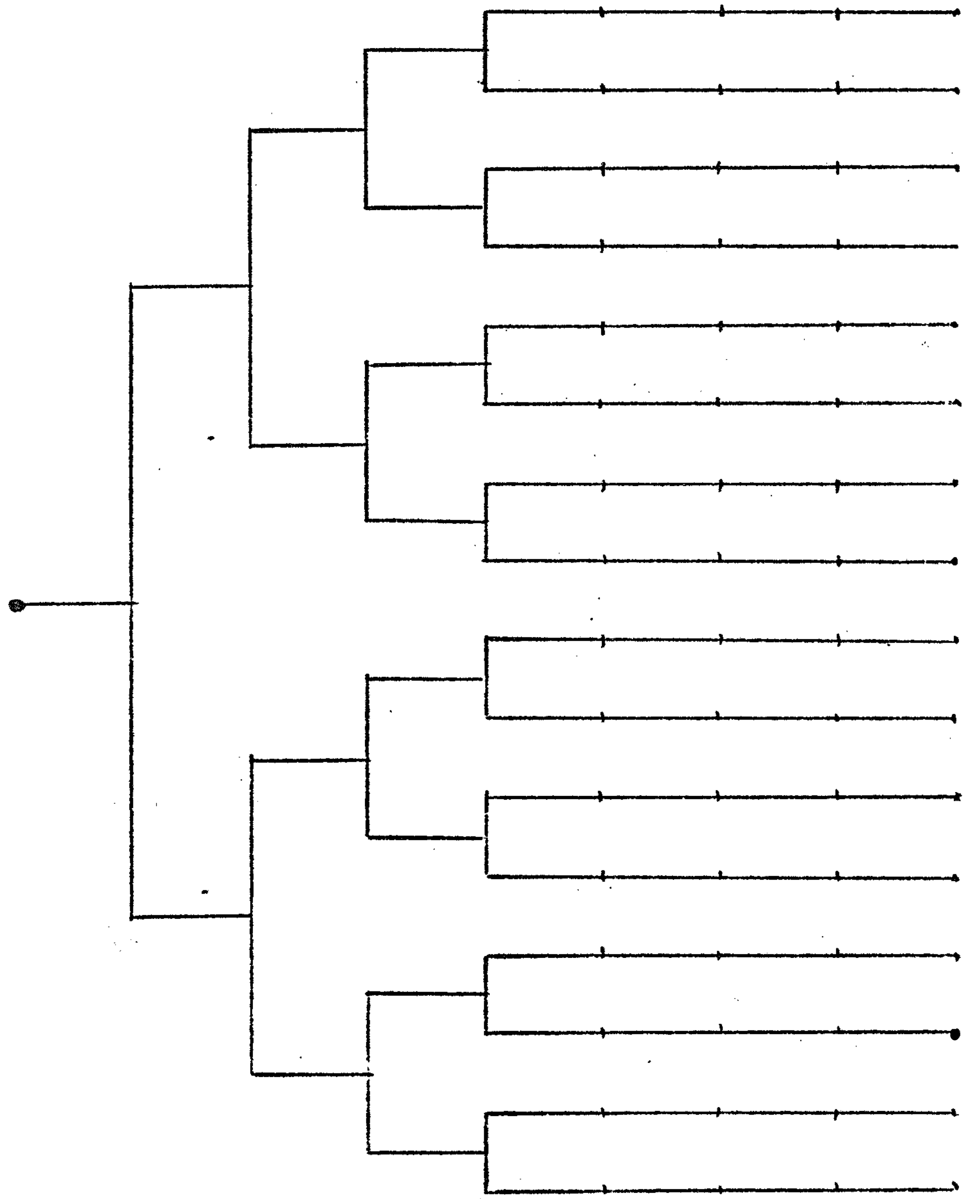


Figure 8 Sample Response Functions

Fig. 1



$\vdash$ $K = 4$ $\rightarrow\mid\leftarrow$ $t = 3$ $\rightarrow\mid$

FIG 2 : terminated code tree

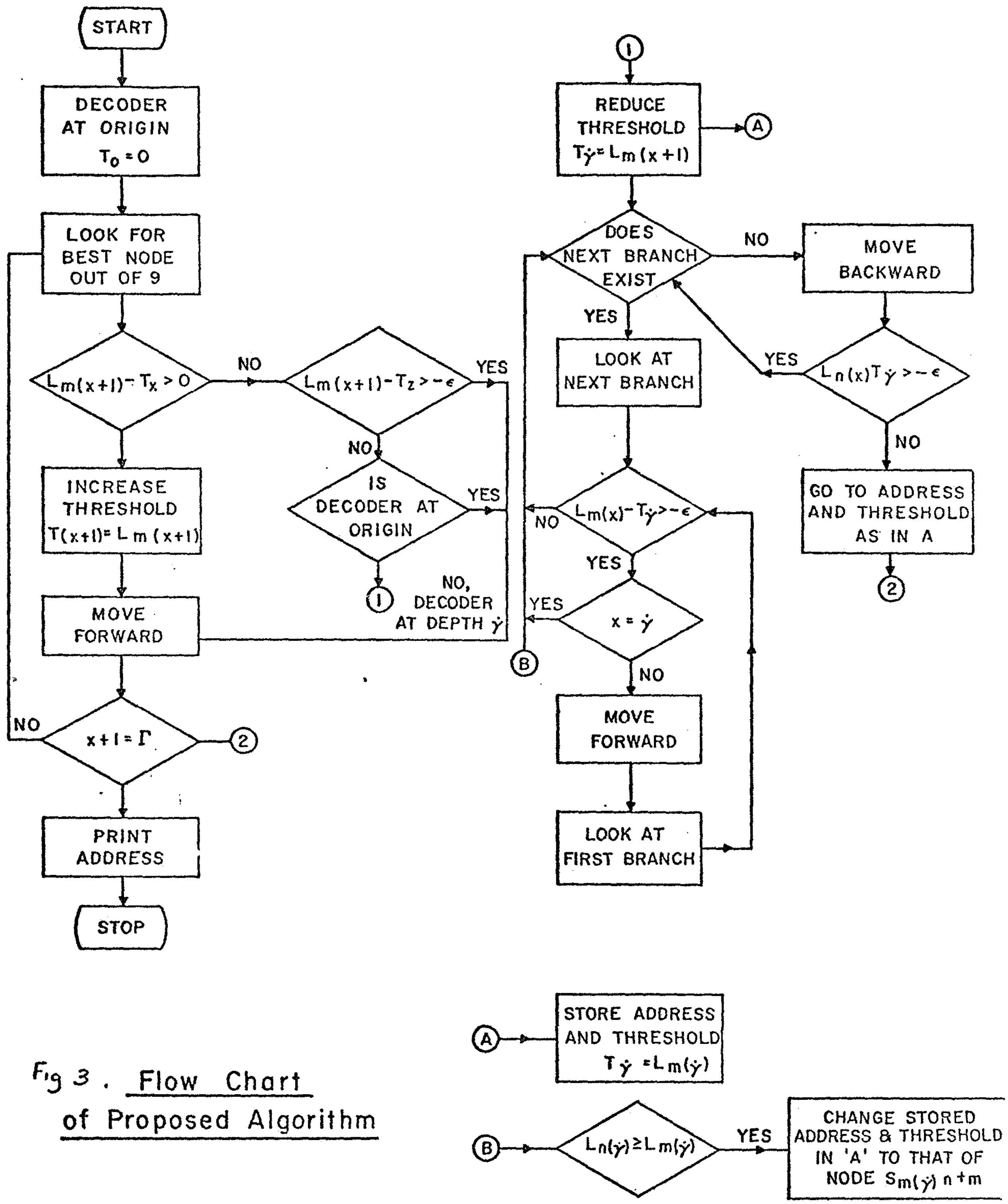


Fig 3. Flow Chart of Proposed Algorithm