

Cyclic Equalization—A New Rapidly Converging Equalization Technique for Synchronous Data Communication

By K. H. MUELLER and D. A. SPAULDING

(Manuscript received June 6, 1974)

A new technique for very fast start-up of adaptive transversal-filter equalizers used in high-speed synchronous data communications is presented. A special training sequence whose period in symbols is equal to the number of equalizer taps is used initially to achieve an open eye pattern. Rapid convergence, even over highly distorted channels, is obtained because an ideal reference sequence is available at the receiver, but it is not necessary to synchronize the ideal reference with the received sequence. The special choice of the training sequence automatically provides the synchronized ideal reference needed for fast convergence, but the resulting equalizer coefficients may be cyclically displaced from their proper positions. After the eye is opened by this process, the equalizer coefficients are rotated to their proper positions, and decision-directed equalization is used with either a longer training sequence or random data to achieve final tap settings. Adjustments during the training period can be made with a gradient-type algorithm or with stochastic adjustment techniques; an exact analysis is possible for both of these schemes. Cyclic equalization is shown to provide perfect equalization at evenly spaced points in the frequency domain.

I. INTRODUCTION

The effective data throughput in polling systems is, to a large degree, dependent on the start-up time of the data modems that are used in the network. Many of these systems operate at high speed and transmit data blocks of comparatively short duration. At 4800 b/s, a 1000-bit block is transmitted in about 200 ms, and to achieve a reasonable overall efficiency, the time needed to condition the modem for transmission (start-up) should be short in comparison to the time required to transmit an average block. This becomes increasingly difficult with

higher modem speeds. Prior to the transmission of the actual data, timing and carrier information must be recovered very accurately, and the adaptive equalizer that is necessary to cope with the linear channel distortion at such high speeds must be trained.

The time required to adjust the equalizer represents the bulk of the modem start-up time; it is thus important to study in detail the problems associated with fast equalizer start-up. The most common structure of such an equalizer consists of a transversal filter with a set of controlled gain coefficients that are spaced at the symbol interval T ; and the start-up problem is to find an initial set of "reasonably good" values for these coefficients in a very short time. The purpose of this paper is to present a practical method for doing this.

We first provide some background and discuss some factors that affect equalizer start-up. This leads to the principle of cyclic equalization that is discussed in Section III. Sections IV through VIII discuss the operation of the cyclic equalizer using the mean-square tap-adjustment algorithm where averaging is used to compute the adjustment signals. The optimum tap coefficients are discussed and shown to provide perfect equalization of the channel at equally spaced points in the frequency domain. The relationship is explained between the eigenvalues of the channel correlation matrix, which control the convergence of the adaptive algorithm, and the discrete Fourier transform of the received training signal. Selection of the training sequence and the starting values of the tap coefficients and the effects of noise are also discussed. Finally, in the remaining sections, a more practical implementation is analyzed of the cyclic equalizer that does not use averaging in the tap adjustment algorithm (stochastic adjustment). The analysis of this algorithm is, in general, very difficult but, in the case of the cyclic equalizer, the time-varying difference equation that describes the noiseless equalization process can be solved exactly, and the conditions for this algorithm to be stable can be developed. Again, here the stability of the algorithm is related to the discrete Fourier transform (DFT) of the received signal. It is shown that the algorithm converges if the DFT of the received signal has no zero elements—that is, if the received signal spectrum has no nulls. This material along with a brief discussion of the asymptotic behavior of the algorithm is given in Sections IX through XII.

Within the paper, we also discuss various implementations, including a method to further speed up the tap calculation using an accelerated signal-processing technique. It will be seen that cyclic equalization is very attractive and economical to implement. Actual convergence is presented with some computer simulations that demonstrate the fast start-up capabilities of the new method.

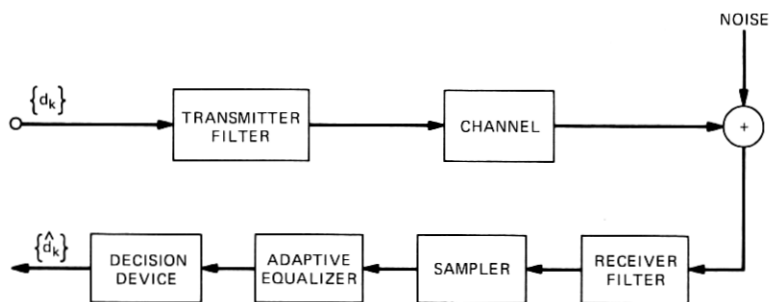


Fig. 1—Block diagram of data transmission system.

II. BACKGROUND

We will consider the pulse-amplitude modulated data system shown in Fig. 1. Data symbols, d_k , are transmitted every T seconds through a transmitter low-pass filter. This signal then passes through a distorting channel that has been made baseband by the modulation-demodulation process inherent in the modem, noise is added, and the composite signal is sampled every T seconds after the receiver filter. The sampled signal vector \mathbf{x}_k is then equalized by a transversal filter with coefficient vector \mathbf{c} (see Fig. 2) to produce an output $y_k = \mathbf{x}_k^T \mathbf{c}$ upon which the decision device operates to produce estimates, \hat{d}_k , of the transmitted symbols. The receiver structure has the form of the optimum linear receiver⁶ but, because the channel is never precisely known and changes with time, the transversal equalizer is made adaptive to optimize performance.

Our concern in this paper is with the equalizer and ways to make it adapt rapidly from some initial setting to its final setting. A large number of papers, a partial list of which has been included in Refs. 1 to 51, have been written about equalizers, algorithms for adjusting

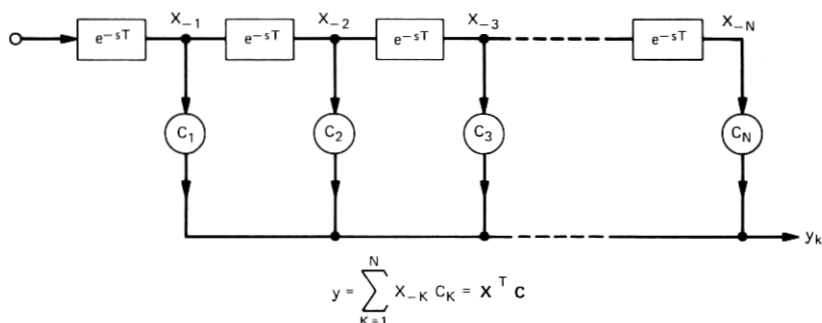


Fig. 2—Nonrecursive transversal filter.

them, and the speed with which these algorithms converge. In developing procedures for adapting the equalizer coefficients, some appropriate performance measure must be defined that will discriminate between good and bad coefficient vectors. Although our goal is to minimize the probability of error, this criterion is too difficult to work with directly. As a result, secondary performance measures such as the peak distortion,²

$$D = h_0^{-1} \sum_{k \neq 0} |h_k|, \quad (1)$$

or the mean-square error,

$$\epsilon = E\{|y_k - d_k|^M\}, \quad M = 2, \quad (2)$$

are used. In (1), h_k is the sampled system impulse response. The peak distortion is related to the "eye opening,"⁶ and for binary symbols and noiseless transmission, $D < 1$ implies no decision errors. In (2), $E\{\cdot\}$ is the expectation operation and $y_k - d_k$ is the remaining error at the equalizer output. These performance measures (M could be greater than 2, if desired) can be shown to be convex functions of the equalizer coefficients, thereby proving the existence of a global minimum.

We will work primarily with the mean-square error (MSE) criterion. This criterion includes the effects of noise, whereas the peak distortion criterion does not, it is convenient to work with mathematically, it can be used to bound the probability of error,^{5,2} and it leads to adaptive algorithms that are easy to implement. Using the MSE, the optimum coefficient vector for the equalizer can be determined easily. Assuming $E\{d_k^2\} = 1$, we have from (2)

$$\epsilon = \mathbf{c}^T \mathbf{A} \mathbf{c} - 2\mathbf{c}^T \mathbf{v} + 1, \quad (3)$$

where

$$\mathbf{A} = E\{\mathbf{x}_k \mathbf{x}_k^T\} \quad (4)$$

is the signal autocorrelation matrix,

$$\mathbf{v} = E\{d_k \mathbf{x}_k\} \quad (5)$$

defines the signal correlation vector, and \mathbf{x}_k is the vector of tap signals at the k th time instant. Finding the gradient of (3) with respect to the tap gains gives

$$\mathbf{g} = 2E\{(y_k - d_k)\mathbf{x}_k\} = 2(\mathbf{A}\mathbf{c} - \mathbf{v}). \quad (6)$$

Our optimization problem has a unique solution if \mathbf{A}^{-1} exists. Setting (6) equal to zero yields

$$\mathbf{c}_{\text{opt}} = \mathbf{A}^{-1}\mathbf{v} \quad (7)$$

$$\epsilon_{\text{opt}} = 1 - \mathbf{v}^T \mathbf{c}_{\text{opt}} = 1 - \mathbf{v}^T \mathbf{A}^{-1} \mathbf{v}. \quad (8)$$

The problem of equalizer start-up is simply to find the solution to (7) in a rapid and economical manner. The economical part of the question is very important. One can imagine a start-up procedure that operates by sending a special training signal for a short period of time. The received signal, $x(t)$, is stored at the receiver. The training sequence is known at the receiver, but its absolute time reference is not known. The receiver contains a very fast high-power computer which now, in essentially no time, computes (8) for a large number of different time references and finds the time reference for the locally stored training sequence that minimizes ϵ_{opt} . The computer has thus accomplished both synchronization and equalization. This hypothetical system achieves a start-up time limited only by the time required to transmit the training signal but, with today's technology, its speed-cost product, if you will, is very poor. It does not represent an economical solution to the problem. Many currently proposed fast start-up equalizers, although not as extreme as this example, still do not present cost-effective solutions.

In addition to the economic aspect, this example illustrates two other important points. The first is the solution of (7). Much of the work on equalization is concerned with efficient algorithms that avoid the direct matrix inversion and obtain an iterative solution. Often, however, the time required to perform the calculations in (4) and (5) is not explicitly considered in evaluating start-up time. Second, synchronizing the stored reference signal in the receiver can take significant time, and that aspect of start-up time seems to be universally ignored.

Now we consider the solution of (7) in more practical terms. A well-known approach for solving (7) is

$$\mathbf{c}_{m+1} = \mathbf{c}_m - \beta_m (\mathbf{A}\mathbf{c}_m - \mathbf{v}), \quad (9)$$

i.e., a first-order steepest-descent gradient algorithm. For appropriate conditions on β_m , \mathbf{c}_m converges to \mathbf{c}_{opt} .

According to (6), the gradient is obtained by correlating the tap-signal vector and the error voltage

$$\mathbf{g} = 2E\{e_k \mathbf{x}_k\}. \quad (10)$$

From an implementation point of view, this is a convenient quantity because the signal vector, \mathbf{x}_k , is readily available, and the error, $e_k = y_k - d_k$, can be estimated. A difficulty still remains in that the expected value is not available in real time and must be estimated by averaging over a finite number of symbols. The difference equation (9) then takes the form

$$\left. \begin{aligned} \mathbf{c}_{m+1} &= \mathbf{c}_m - \beta_m \cdot \frac{1}{L} \sum_{k=mL}^{mL+L-1} \mathbf{x}_k (\mathbf{x}_k^T \mathbf{c}_m - d_k) \\ &= \mathbf{c}_m - \beta_m (\mathbf{A}_m \mathbf{c}_m - \mathbf{v}_m) \end{aligned} \right\} \quad (11)$$

Averaging is done over L symbols between succeeding adjustments. If random data are transmitted, A_m and \mathbf{v}_m will depend on the particular signal pattern of each iteration interval and are random variables with mean A and \mathbf{v} and variances decreasing with longer averaging interval L . The analysis of the behavior of (11) is difficult, particularly when we try to determine ways of improving the convergence rate. By reducing L , we can make many more iterations in a given time, but we must use a smaller β -value to take into account the larger variance of the calculated gradient. Longer averaging between each step would take more time but give a better estimate for the gradient, and therefore allow a somewhat higher value of β . Mosen¹³ has studied the optimization of the averaging interval, assuming an ideal reference and Gaussian signals. In this special case, the optimum value is $L = 1$; i.e., corrections are made after each symbol and no averaging at all is done. This method is often called "stochastic approximation," because the corrections are stochastic quantities whose means equal the desired gradient.

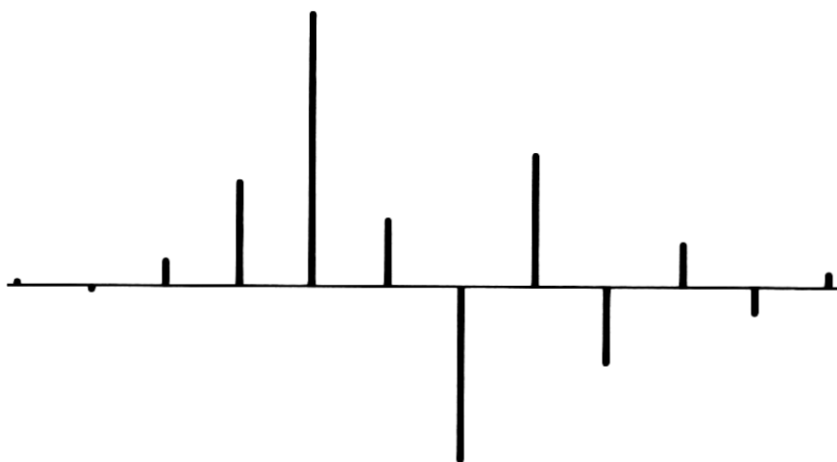
At this point, it appears that the mean-square algorithm with no averaging, i.e.,

$$\left. \begin{aligned} \mathbf{c}_{m+1} &= \mathbf{c}_m - \beta_m e_m \mathbf{x}_m \\ &= (\mathbf{I} - \beta_m \mathbf{x}_m \mathbf{x}_m^T) \mathbf{c}_m + \beta_m d_m \mathbf{x}_m \end{aligned} \right\}, \quad (12)$$

is an attractive scheme to investigate further to obtain fast real-time convergence. There remain, for the moment, two main difficulties that need further discussion. The first one is the problem of obtaining the data values d_k . They can be estimated in the usual way from a threshold decision, but on channels with large distortion the initial error rate may be close to 50 percent and estimates \hat{d}_k are very unreliable in such a situation. An algorithm with a decision-directed reference may thus behave erratically, and convergence cannot be guaranteed. The results of a few simulations will give some further insight.

The channel assumed for the simulation consisted of a 10-percent cosine roll-off baseband filter with parabolic delay distortion [$5.4T$ at the Nyquist frequency ($1/2T$)] and a sampling offset of $0.3T$ from the peak of the response. The resulting channel response from a single pulse is shown in Fig. 3. This same pulse was also used by Hirsch and Wolf.¹¹ The initial peak distortion is 2.62, resulting in a completely closed eye pattern.

The first simulation is for the algorithm (12), but an estimated reference (obtained from a threshold decision) was used. Figure 4 shows the resulting peak distortion versus the number of symbols for four different step sizes. Decision errors are responsible for random distortion increases rather than reductions. This is avoided in Fig. 5



PULSE RESPONSE
 10% COSINE ROLL-OFF BASEBAND FILTER WITH PARABOLIC DELAY DISTORTION
 TIMING OFFSET FROM PEAK RESPONSE : 30%

Fig. 3—Impulse response with peak distortion of 2.62.

where we have repeated the same runs with an ideal reference signal. The improvement is significant. Note that the ideal reference signal is really needed only until the peak distortion has decreased sufficiently to yield an open eye pattern; from this time on, the error probability is essentially zero, and a decision-directed reference can be used.

The difficulty in providing an ideal reference signal lies in the synchronization problem. Remember that we require such a signal only in channels with very large amounts of distortion, but achieving reliable synchronization in the presence of severe distortion is a problem in itself that usually calls for time-consuming correlation methods.⁵

A second problem is associated with the choice of the training sequence. Obviously, a strictly random data pattern would be a bad choice, since transitions would only occur on a probabilistic basis and not be guaranteed. The variability of repeated convergence runs would be large. This can be avoided by transmitting a short-period training sequence. Even if the starting point occurred at random, convergence would be more predictable. We know that we cannot make the period of the training sequence shorter than the duration of the impulse response of the equalizer; otherwise, the tap signals would not be linearly independent, the correlation matrix A would be singular, and a unique solution for the optimum tap vector c would not exist. We will, however, study the limiting case where the period, in symbols, of the training sequence is equal to the number of taps on the equalizer. This will lead directly to the idea of the cyclic equalization. Before we

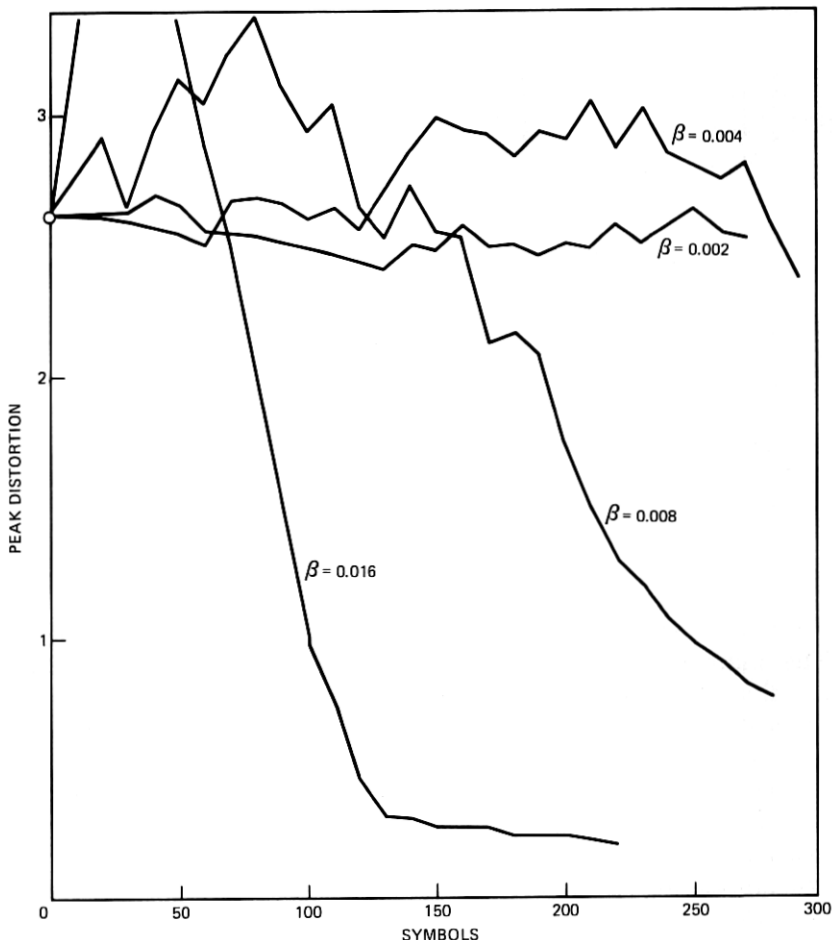


Fig. 4—Convergence behavior of stochastic adjustment algorithm (12) with a decision-directed reference.

do this, we will provide some additional insight by a short discussion of the frequency domain aspects of the equalization problem.

III. PRINCIPLE AND IMPLEMENTATION OF CYCLIC EQUALIZATION

Let the spectrum of the received data signal be $G(\omega)$ and assume that this signal is applied to an N tap equalizer with coefficients c_n , $n = 0, \dots, N - 1$. The resulting output spectrum is

$$X(\omega) = G(\omega) \sum_{n=0}^{N-1} c_n \exp(-j\omega nT), \quad (13)$$

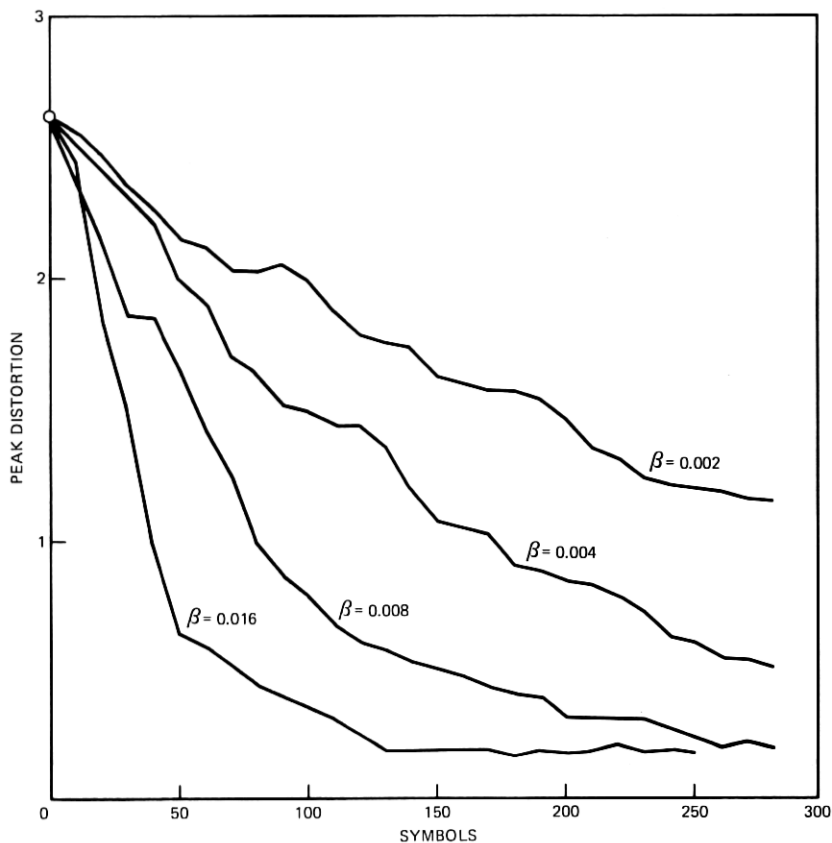


Fig. 5—Convergence behavior of stochastic adjustment algorithm (12) with an ideal reference.

and the overall system would be distortion-free (Nyquist criterion) if

$$\sum_k X \left(\omega + \frac{2\pi k}{T} \right) = \exp(-j\omega\tau), \quad |\omega| < \frac{\pi}{T}. \quad (14)$$

Combination of (13) and (14) yields the condition

$$\sum_{n=0}^{N-1} c_n \exp(-j\omega nT) \sum_k G \left(\omega + \frac{2\pi k}{T} \right) = \exp(-j\omega\tau), \quad |\omega| < \frac{\pi}{T}. \quad (15)$$

Obviously, (15) cannot be satisfied for a finite N and an arbitrary $G(\omega)$. Usually, the coefficients c_n are chosen according to a minimum mean-square-error (MMSE) criterion in the time domain, which is equivalent to an MMSE criterion of (15) in the frequency domain. The problems of such an approach have been discussed in Section II, and we have seen that the commonly used iterative search schemes can be

very efficient during the tracking mode, but initial training may not be without problems.

A closer look at (15) shows that the left-hand side is a linear combination of the coefficients c_n . Although perfect equalization cannot be achieved at all frequencies, it is possible to obtain zero error at a number of specified frequencies ω_m within the range $|\omega_m| < \pi/T$. This is, of course, also true with an MMSE approach, since the resulting transfer function will oscillate around the desired one; i.e., the error will ripple between positive and negative values. The crossing frequencies are, however, not known and usually not of interest. In the new scheme we propose here, we will do exactly the contrary: We will precisely specify the crossing frequencies, although we realize that such an approach will, in general, not yield MMSE. Specifying the frequencies ω_m where perfect equalization is obtained will transform the condition (15) in a set of linear equations for the coefficients c_n . Obviously, we have to consider two cases:

- (i) $N = \text{even}$: $N/2$ different frequencies $\omega_m \neq 0$ must be specified.
- (ii) $N = \text{odd}$: $(N - 1)/2$ different frequencies $\omega_m \neq 0$ and $\omega = 0$ must be specified to obtain a unique solution for the c_n 's.

Theoretically, a set of reference tones ω_m could be transmitted, $G(\omega_m)$ measured at the receiver, and the coefficients computed from (15). Fortunately, it is possible to propose a much more attractive solution.

The generation of the reference tones can be accomplished in a straightforward way if we select the frequencies ω_m equally spaced across the Nyquist band; a suitable periodic data sequence of length NT will produce such spectral lines at $\omega_m = 2\pi m/NT$. Note that the number of symbols in such a training sequence is equal to the number of taps of the equalizer. This choice is extremely important and provides a number of unique advantages to achieve fast equalizer start-up.

We now discuss in detail such a training procedure. Assume an equalizer where an ideal reference signal is used and the period of the training sequence is equal to the number of taps on the equalizer. Assume for the moment that the channel is distortionless and the ideal reference is synchronized with the incoming signal. If we let the adaptive algorithm adjust the equalizer taps, the center tap on the equalizer will become unity, and all the others will be zero. This is really what we mean when we say the reference is synchronized; that is, the optimum equalizer coefficients are centered on the equalizer rather than shifted off to one end or the other. Now again, for this "ideal" example, if the reference signal is delayed by one symbol from perfect synchronization, the adaptive algorithm will cause the equalizer coefficient one position removed from the center to become unity, and all the

others will be zero. The movement of the unity gain tap by one position indicates a one-symbol delay in synchronization of the ideal reference. In an actual situation, the other taps on the equalizer will be nonzero and, with an unsynchronized reference, the adaptive algorithm will cause tap coefficients to occur that are cyclically rotated from those that would occur if the reference were synchronized.

To say this another way, if the training sequence is periodic with a period equal in symbols to the number of taps of the equalizer, the received signal is then also periodic (neglecting noise effects), and one full period of the sequence is always stored in the equalizer. Each symbol that is shifted out at the end of the delay line is replaced by an identical new symbol at the input. This is more clearly shown in Fig. 6 for a seven-tap equalizer with taps c_1 through c_7 and a seven-bit sequence x_1 through x_7 . At time $t_0 + 2T$, it is seen that the stored sequence has been cyclically shifted by two units as compared to the time t_0 . But it is also seen that the same output signal $y(t_0 + 2T)$ could have been obtained at time $t = t_0$ if the taps were cyclically shifted back by two positions. Thus, at any given time $t = t_0$, all outputs $y(t = t_0 + kT)$ can be obtained with a suitable cyclic shift of the components of the tap vector.

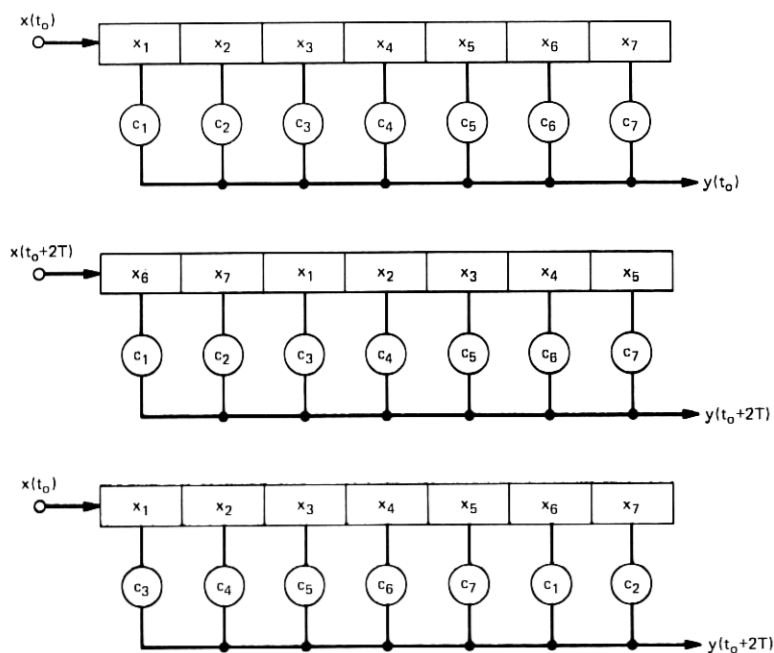


Fig. 6—Basic idea of cyclic equalization.

This feature provides an elegant solution to the synchronization problem. Any cyclic shift between the received sequence and the reference sequence will yield a compensating cyclic shift of the (same) tap coefficients. It is, therefore, not necessary to achieve synchronization prior to equalization, but it is of course necessary to properly shift the tap coefficients after initial training to prepare the equalizer for random data. This can easily be done by cycling them in such a way that the largest coefficient is aligned with a reference position, e.g., the center tap. Because of its particular features just described, we will call this novel start-up scheme "cyclic equalization."^{42,43}

The possible structure of such an equalizer is outlined in Fig. 7. An internal word generator produces an ideal reference sequence that need not be synchronized with the received sequence. All taps are initially preset to identical values (since the location of the "center tap" is not known). The equalizer will then produce a set of taps with the particular cyclic shift corresponding to the "synchronization delay." After this initial training, the tap coefficients are cyclically shifted for "alignment," as outlined above. At this point, the equalizer has reasonably good tap coefficient settings and the peak distortion at the output is less than unity; i.e., the eye is open and, in the absence

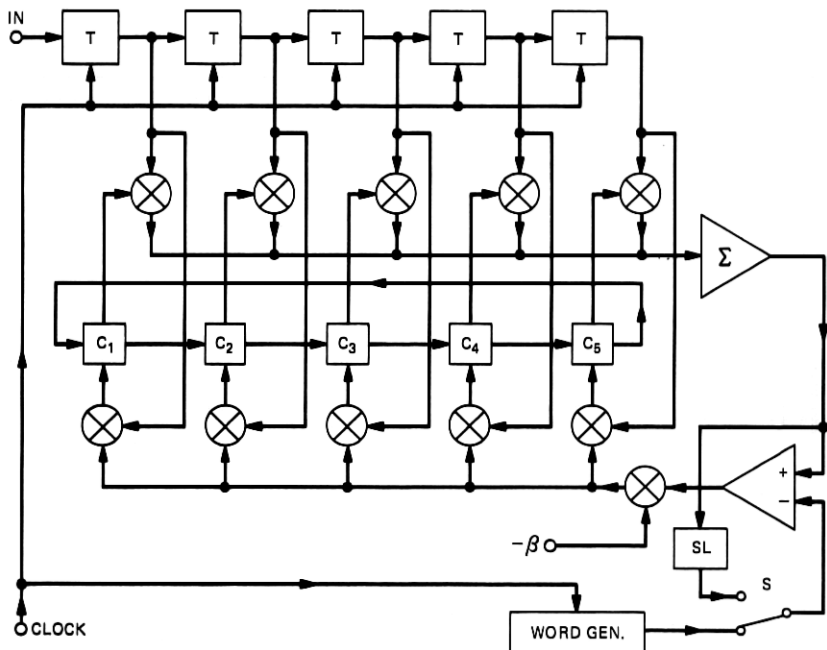


Fig. 7—Block diagram of an equalizer with cyclic start-up.

of noise, errorless decisions can be made. Fast coarse adjustment of the tap coefficients has been achieved without wasting time synchronizing the ideal reference. Once the eye is open, decision-directed equalization can be used with a somewhat longer training sequence or random data to achieve the final fine adjustment of the tap coefficients.

The fact that mean-square equalization with a training sequence period equal to the length of the equalizer can give very fast and very consistent, relative to the starting point of the adaptation, equalization has been demonstrated in numerous simulations. One of these is illustrated in Fig. 8. The same channel is used for this example as was used previously; the peak distortion is 2.62, the signal-to-noise ratio is 30 dB, and the step size is 0.04. In this case, the equalizer has 15 taps and a 15-bit maximum length training sequence is used because of its nice spectral properties. Adjustments are made at the symbol

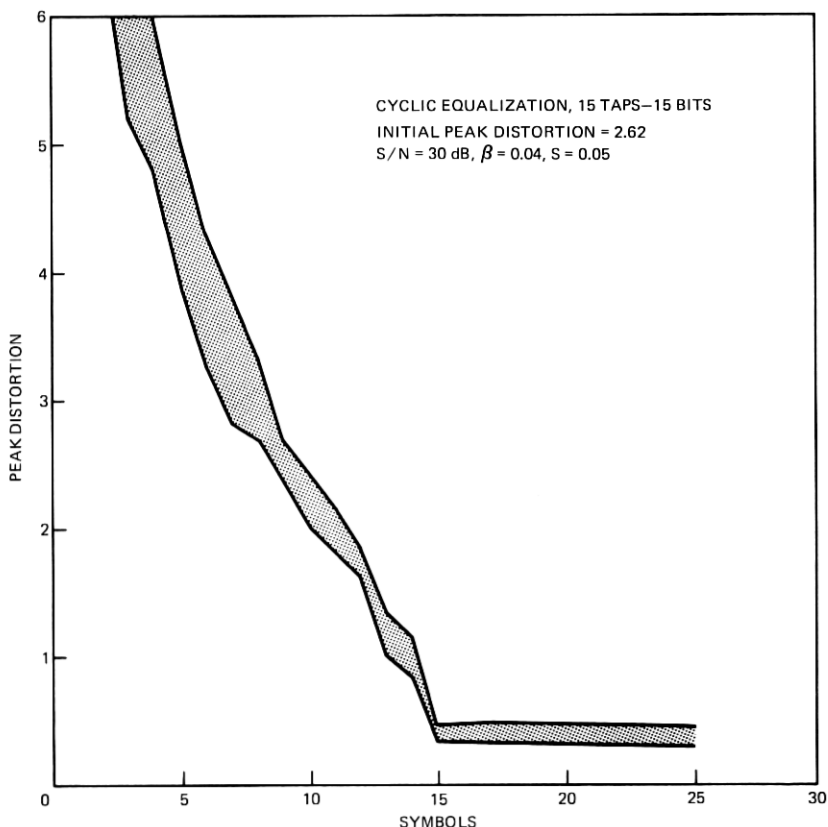


Fig. 8—Start-up behavior with cyclic equalization.

rate. The shaded region in the figure contains all 15 possible convergence curves that correspond to the different starting points for adaptation. Not only are all the convergence curves very similar, but they all achieve a peak distortion of about 0.4 or less in 15 symbols.

A few words are in order about the presetting of the tap coefficients. Because an unsynchronized reference is used, the location of the largest coefficient is *a priori* unknown. It is therefore reasonable to preset all coefficients to identical initial values s , as we have already mentioned above. With most channels, tap coefficients of both polarities will evolve so that one might consider setting $s = 0$ for an unknown channel. The large final value of the center tap would, however, suggest that slightly biased initial conditions might give faster convergence; we will make more precise statements about that in Section VII.

The discussed method of presetting has, of course, some consequences if a channel with low distortion or even an ideal channel were used. In such a situation, a conventional equalizer could do a better job because it would be started with the optimum tap settings ($c_k = \delta_{0k}$) right away and need not make any corrections at all. The cyclic equalizer would have to "converge" even with an ideal input signal; simulations of this case have shown a convergence plot similar to that of Fig. 8.

As a final example, we present the results of a VSB system that is operated over a channel with "parabolic-like" delay (exponent = 2.73) and an s/n of 30 dB. The received and demodulated signal is sampled with different timing phases spaced $T/4$ apart and equalized in a cyclic equalizer with $N = 15$ taps. The distortion values resulting after equalization during only one sequence (i.e., 15 symbols) are summarized in Table I. For comparison, the initial channel distortion D_{Channel} and the minimum distortion D_{min} that can be achieved with an equalizer of this length are also included. It can be seen that initial training using cyclic equalization achieves a performance that is already close to optimum.

Some comments should be made about the simulation results we have presented. They indicate that initial training with cyclic equalization may only be necessary for a very short time; in some cases, for only one sequence period. This means that the received signal is not

Table I — Distortion for VSB channel and 15-tap equalizer

Timing	D_{Channel}	$D_{\text{Cycl.}}$	D_{min}
0	2.04	0.15	0.06
25%	1.87	0.52	0.21
50%	2.25	0.99	0.98
75%	2.88	0.25	0.12

really periodic and that no spectral lines in the strict sense will occur at equally spaced frequencies ω_m , as we specified earlier in this section. The spectrum will be continuous, showing increasingly concentrated peaks at those frequencies with larger numbers of sequence repetitions. We have not found this to be a disadvantage; in fact, under some circumstances, the tap settings achieved with only a small number of iterations were, for the transmission of random data, preferable to the steady-state solution.

We have shown by example that fast reliable initial convergences can be achieved using an ideal reference signal without spending any time to synchronize the reference. Final fine adjustment of the taps is accomplished in a decision-directed mode using a longer sequence or random data. In the next sections, we will analyze the behavior of the cyclic equalizer during its initial training period. The convergence behavior with the mean-square algorithm with averaging, the choice of the training sequence, and the effect of the initial value of the taps will be considered. Then the exact behavior of the mean-square algorithm without averaging will be analyzed and conditions for convergence will be given.

IV. STEADY-STATE SOLUTION FOR THE TAPS

As was discussed in the previous section, the operation of the cyclic equalizer does not depend upon the synchronization of the reference, and we will not stress the rotation property of the taps unless necessary.

We assume a system with N equalizer taps and let the samples of the received signal be the components of the vector

$$\mathbf{x}^T = (\gamma_N, \gamma_{N-1}, \dots, \gamma_1). \quad (16)$$

If we neglect the noise components, the tap-signal vector is periodic and successive vectors are cyclic shifts of each other ($\gamma_{N+m} = \gamma_m$). We define a signal matrix

$$S = \begin{pmatrix} \gamma_N & \gamma_{N-1} & \gamma_{N-2} & \dots & \gamma_1 \\ \gamma_1 & \gamma_N & \gamma_{N-1} & \dots & \gamma_2 \\ \gamma_2 & \gamma_1 & \gamma_N & \dots & \gamma_3 \\ \vdots & \vdots & \vdots & \dots & \vdots \\ \gamma_{N-1} & \gamma_{N-2} & \gamma_{N-3} & \dots & \gamma_N \end{pmatrix}, \quad (17)$$

whose rows consist of all N succeeding sample vectors. The elements of S are given by

$$s_{ik} = \gamma_{(i-k) \text{ Mod } N}. \quad (18)$$

At the equalizer output, a sequence of values $\mathbf{x}^T \mathbf{c}$ (\mathbf{c} is the tap-weight vector) appears as the input vector \mathbf{x} is cyclically shifted through its

N states. The resulting output sequence is

$$\mathbf{y} = S\mathbf{c}. \quad (19)$$

Obviously, it is possible to obtain from a given input sequence any arbitrary desired output sequence by a suitable choice of \mathbf{c} , provided only that S^{-1} exists. If we define a data vector ξ which contains the reference values associated with \mathbf{y} , it is possible to select \mathbf{c} so that

$$\mathbf{y} = \xi = S\mathbf{c}, \quad (20)$$

i.e., the recovered sequence can be perfectly equalized (at least, at the sample points) and there is no residual error. This is even true with nonlinear distortion. Since the error can be reduced to zero, we conclude that the same tap vector

$$\mathbf{c}_0 = S^{-1}\xi \quad (21)$$

is obtained with any equalizer in the steady state, regardless of the particular tap-updating algorithm (as long as it is unbiased).

We now proceed to determine the eigenvalues of the circulant matrix S . Let us first define a set of values r so that

$$r^N = 1 \rightarrow r_k = \exp\left(j \frac{2\pi k}{N}\right). \quad (22)$$

In the next step we form

$$\begin{aligned} \lambda &= \gamma_N + r\gamma_{N-1} + r^2\gamma_{N-2} + \cdots + r^{N-1}\gamma_1 \\ r\lambda &= \gamma_1 + r\gamma_N + r^2\gamma_{N-1} + \cdots + r^{N-1}\gamma_2 \\ &\vdots \\ r^{N-1}\lambda &= \gamma_{N-1} + r\gamma_{N-2} + r^2\gamma_{N-3} + \cdots + r^{N-1}\gamma_N. \end{aligned}$$

This may be written in matrix form as

$$\mathbf{r}_k \lambda_k = S\mathbf{r}_k, \quad (23)$$

where we have defined the vector

$$\mathbf{r}_k = \{r_{kn}\}; \quad \text{with } r_{kn} = \frac{1}{\sqrt{N}} \exp\left(j \frac{2\pi}{N} nk\right). \quad (24)$$

The \mathbf{r}_k 's are obviously eigenvectors of S . The eigenvalues λ_k are

$$\lambda_k = \mathbf{x}^T \mathbf{r}_k, \quad 0 \leq k \leq N-1, \quad (25)$$

and are given by the discrete Fourier transform (DFT) of the input vector \mathbf{x} . The signal matrix S can be diagonalized if we introduce a matrix W with

$$\{W\}_{ik} = \frac{1}{\sqrt{N}} \exp\left(j \frac{2\pi}{N} ik\right), \quad (26)$$

whose columns are made up from the vectors \mathbf{r}_k . W is also symmetric and unitary; the properties

$$W = W^T, \quad W^* = W^\dagger, \quad W^\dagger W = I \quad (27)$$

are easily established. We may now alternatively either express the eigenvalues as components of the diagonal matrix D

$$D = W^\dagger S W \quad (28)$$

or as components of a vector

$$\lambda = W \mathbf{x}, \quad (29)$$

since multiplication with W transforms a vector into its DFT.

We now give an interpretation in the frequency domain. The received (periodic) sequence can be expanded into a Fourier series

$$x(t) = \frac{1}{\sqrt{N}} \sum_m x_m \exp\left(j \frac{2\pi}{NT} mt\right), \quad (30)$$

where the coefficients x_m correspond to the spectral lines and the range of m is determined by the bandwidth. The components γ_i in \mathbf{x} are given by $x(t = \tau + iT)$; this may be combined with (25), and we obtain for the eigenvalues the frequency domain representation

$$\lambda_k = \sum_l x_{k+lN} \exp\left[j \frac{2\pi\tau}{T} \left(l + \frac{k}{N}\right)\right]. \quad (31)$$

In the case where all spectral lines are contained within twice the Nyquist frequency and $\tau = 0$, we have

$$\left. \begin{aligned} \lambda_0 &= x_{-N} + x_0 + x_N \\ \lambda_1 &= x_1 + x_{1-N} \\ \lambda_2 &= x_2 + x_{2-N} \\ &\vdots \\ \lambda_{N-2} &= x_{N-2} + x_{-2} \\ \lambda_{N-1} &= x_{N-1} + x_{-1} \end{aligned} \right\}. \quad (32)$$

As this represents 100-percent excess bandwidth, we may assume that most practical systems are within this range. If spectral lines are only within the Nyquist limit, (32) is simplified to

$$\left. \begin{aligned} \lambda_k &= x_k \quad \text{if } |k| < \frac{N}{2} \\ \lambda_k &= x_{N-k} \quad \text{if } |k| \geq \frac{N}{2} \end{aligned} \right\}. \quad (33)$$

We can now give some comments as to the nature of the resulting tap vector \mathbf{c}_0 . By combining (21) and (28), \mathbf{c}_0 may be expressed as

$$\mathbf{c}_0 = W^{\dagger} D^{-1} W \xi. \quad (34)$$

Here the term $W\xi$ is the DFT of the ideal samples and establishes a set of reference values at equally spaced points in the frequency domain (discrete Nyquist equivalence). The multiplication with D^{-1} determines the gain of an ideal correction function at these points. The resulting tap-vector \mathbf{c}_0 is the inverse DFT of this correction function. The overall transfer characteristic (channel and equalizer) is discrete Nyquist equivalent when $\mathbf{c}_0 = S^{-1}\xi$, i.e., frequency-domain equalization is precise at a set of equidistant points [spacing $(2\pi/NT)$]. This tap vector is, in the general case, not optimum for random data transmission after the training period. Basically, equalization is a mathematical approximation problem. The equalizer approximates the compensation function with a trigonometric polynomial. With a cyclic equalizer, the coefficients are selected to match the desired function at equidistant points. This will generally not give minimum mean-square error at the output, since only discrete frequency information is used and the channel behavior between the sample points is not taken into account. In a recent paper, Chang and Ho⁴⁰ briefly discussed this problem from a somewhat different point of view and concluded that the initial approximation \mathbf{c}_0 is generally close to the optimum settings for random data. We will not further discuss the approximation problem in this paper.

V. MEAN-SQUARE ALGORITHM WITH AVERAGING

In this section, we are looking at a tap-control system that minimizes the mean-square error between the equalizer output $y(nT)$ and reference symbols ξ_n . We use a steepest descent gradient algorithm of the form

$$\mathbf{c}_{m+1} = \mathbf{c}_m - \beta(A\mathbf{c}_m - \mathbf{v}), \quad (35)$$

where A is the signal-correlation matrix and \mathbf{v} is the signal-correlation vector. The gradient

$$\mathbf{g}_m = E\{\mathbf{x}_i(y_i - \xi_i)\} |_{\mathbf{c}=\mathbf{c}_m} \quad (36)$$

is evaluated in the usual way by time averaging.

First we note that, in the noiseless case, because of the cyclic nature of \mathbf{x} , both A and \mathbf{v} can be determined by time-averaging over one full sequence length of N symbols. Further, A and \mathbf{v} are constant and well-defined throughout the process. It is easily verified that A can be

expressed in terms of the normal signal matrix S ,

$$A = \frac{1}{N} SS^{\dagger} = \frac{1}{N} S^{\dagger}S, \quad (37)$$

and that \mathbf{v} is equivalent to

$$\mathbf{v} = \frac{1}{N} S^{\dagger}\xi. \quad (38)$$

The gradient is zero, and updating stops when

$$\mathbf{c} = \mathbf{c}_0 = A^{-1}\mathbf{v} = S^{-1}\xi. \quad (39)$$

If we introduce the tap error vector $\delta_m = \mathbf{c}_m - \mathbf{c}_0$, (35) takes the form

$$\delta_{m+1} = (I - \beta A)^m \delta_1. \quad (40)$$

The choice of β and the convergence depend on the eigenvalues of A . To guarantee that $\delta_{m+1} \rightarrow 0$ for large m , we require $0 < \beta < 2/\mu_{\max}$, where μ_i are the eigenvalues of A . Since

$$S = WDW^{\dagger} \quad (41)$$

and therefore

$$A = \frac{1}{N} SS^{\dagger} = \frac{1}{N} WDD^{\dagger}W^{\dagger}, \quad (42)$$

the eigenvectors of A and S are common (and independent of \mathbf{x}). The eigenvalues μ_k of A are related to the eigenvalues λ_k of S by

$$\mu_k = \frac{1}{N} \lambda_k^* \lambda_k. \quad (43)$$

Another interpretation is obtained by realizing that the matrix A is circulant (like S) and symmetric with elements

$$\{A\}_{ik} = a_{i-k} = a_{k-i} = a_n = \frac{1}{N} \sum_{m=l}^{l+N-1} \gamma_m \gamma_{m-n}. \quad (44)$$

By analogy to (25), the eigenvalues are

$$\mu_k = \mathbf{a}^T \mathbf{r}_k; \quad 0 \leq k \leq N-1, \quad (45)$$

where \mathbf{a} contains the components a_n . We see that the eigenvalues are given by the DFT of the cyclic autocorrelation values, a_n .

We now express these eigenvalues by the spectral lines in the frequency domain. If we combine (31) and (43), we obtain

$$\mu_k = \frac{1}{N} \sum_m \sum_n \chi_{k+mN} \chi_{k+nN}^* \exp \left[j \frac{2\pi\tau}{T} (m-n) \right] \quad (46)$$

or, in an equivalent form,

$$\mu_k = \frac{1}{N} \sum_m \sum_n \chi_{k-mN} \chi_{(n+m)N-k} \exp\left(j2\pi n \frac{\tau}{T}\right). \quad (47)$$

If the bandwidth is Nyquist-limited, only one single term in (47) contributes to the eigenvalues, namely,

$$\mu_k = \mu_{N-k} = \frac{1}{N} |\chi_k|^2, \quad k \leq \frac{N}{2}. \quad (48)$$

The eigenvalues are then independent of timing and carrier parameters and phase distortion of the channel. Only the signaling format, the channel attenuation, and the choice of the sequence ξ determine the eigenvalues. (Note that we have so far not restricted the choice of ξ to a particular class of sequences, such as maximum length sequences.)

In the case of excess bandwidth which is, however, limited to twice the Nyquist frequency (all reasonable pulse-amplitude modulation systems fall in this category), a few more terms in (47) need be considered, and

$$\begin{aligned} \mu_k = & |\chi_k|^2 + |\chi_{N-k}|^2 \\ & + \chi_k \chi_{N-k} \exp\left(j \frac{2\pi\tau}{T}\right) + \chi_k^* \chi_{N-k}^* \exp\left(-j \frac{2\pi\tau}{T}\right), \end{aligned} \quad (49)$$

which shows the influence of the timing phase τ . Note that only the third and fourth terms depend on phase and timing parameters. This term represents the fold-over around the Nyquist frequency.* The smaller the roll-off, the less the eigenvalues will be affected by this fold-over. In fact, it is even possible to have a small amount of excess bandwidth without any contribution of these terms. This is shown in Fig. 9. We distinguish two cases.

- (i) $N = \text{odd}$. The Nyquist frequency is located midway between two spectral lines of the training sequence. Fold-over is avoided if we have a normalized roll-off $\alpha \leq 1/N$.
- (ii) $N = \text{even}$. The Nyquist frequency coincides with a spectral line of the training sequence. If we choose $\alpha \leq 1/N$, the eigenvalues will still be phase-invariant, but one of them (for $k = N/2$) will now be dependent on the timing phase t_0 .

Most voice-grade telephone channels have very large phase distortion, but only moderate amplitude distortion. Usually, the worst-case gain deviations over a given frequency range are known (e.g., on

* Some crossterms in (49) are zero if the bandwidth is less than twice the Nyquist frequency.

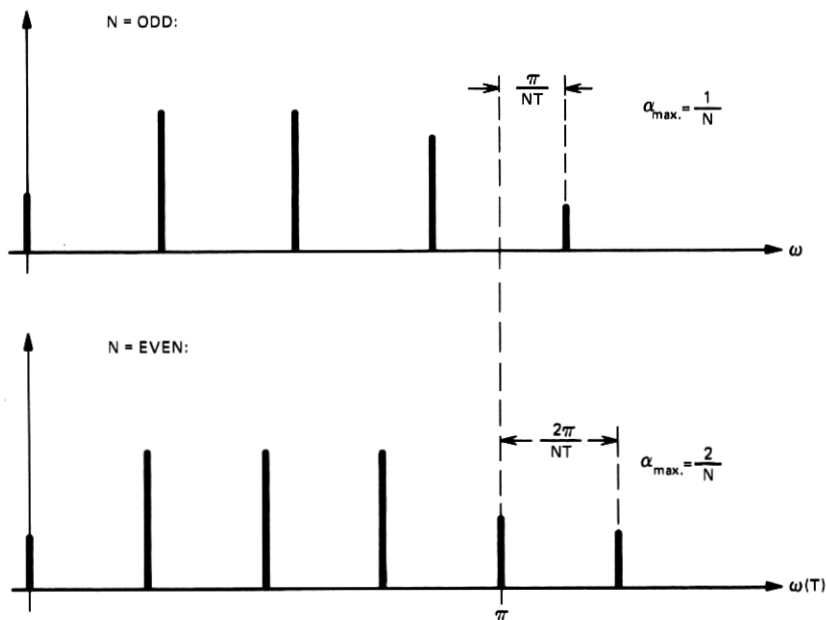


Fig. 9—Spectrum of training sequence.

private channels). If we deal with small excess bandwidth and we know our sequence ξ , it is obviously possible to calculate the spread of the eigenvalues from (46) or (49). We may then choose the value of β in (35) so that

$$0 < \beta < \frac{2}{\lambda_{\max}} \quad (50)$$

to insure convergence. In addition, it may, of course, be necessary to normalize the signal power $\mathbf{x}^T \mathbf{x}$ with an automatic gain control to make the eigenvalues dependent only on the relative gain difference between the various frequencies, but independent of the average absolute gain.

VI. CHOICE OF THE TRAINING SEQUENCE

So far, we have not discussed the choice of the training sequence ξ . From the previous study we know that the eigenvalues of S and A are well-behaved as long as the DFT of \mathbf{x} , or of the sampled autocorrelation function, respectively, has no zero elements. This is obviously sufficient to guarantee the existence of inverses of S and A and therefore also the existence of a solution \mathbf{c}_o . Zero elements can be avoided by selecting a signaling format and a sequence ξ to insure nonzero line amplitudes at all frequencies $f_n = n/NT$ within the transmission bandwidth. If the channel does not have serious attenuation gaps, we have

also nonzero amplitudes at the receiver input. To obtain fast convergence, the eigenvalues should be as equal as possible (minimum spread). This can obviously best be achieved by selecting a sequence ξ which produces lines of equal amplitudes; predistortion for expected attenuation at the band edges is possible. The transmitter is then effectively sending a comb of equally spaced frequencies of approximately equal amplitudes that could obviously also be provided by a number of frequency generators, but are, of course, much more efficiently synthesized as spectral lines of a suitable sequence. Note that the samples of the training sequence ξ need not necessarily be binary; arbitrary numbers (and sequence lengths) can be stored in ROM's in both transmitter and receiver. We will discuss a few special cases for ξ , assuming small excess bandwidth ($<1/N$) and an odd number of taps and flat gain:

- (i) Single pulse, $\xi^T = (0, \dots, 0, 1, 0, \dots, 0)$: This produces a frequency comb of equal amplitudes. We have further

$$A = \frac{1}{N} I, \quad \lambda_k = \text{const} = 1/N, \quad \beta_{\text{opt}} = N. \quad (51)$$

Convergence is obtained in a single iteration, independent of the initial settings. See eq. (40).

- (ii) Single pulse, $\xi^T = (1, \dots, 1, -1, 1, \dots, 1)$: This produces a similar frequency comb, but with a much larger amplitude at dc. The eigenvalues are shown to be

$$\lambda_0 = (N - 2)^2/N; \quad \lambda_1 = \dots = \lambda_{N-1} = 4/N. \quad (52)$$

- (iii) Maximum-length pseudorandom sequence: Such sequences have lengths $N = 2^m - 1$ (among others), and were used for the simulations given earlier in Section II. The eigenvalues are

$$\lambda_0 = \frac{1}{N}; \quad \lambda_1 = \dots = \lambda_{N-1} = 1 + \frac{1}{N}. \quad (53)$$

For a given symbol magnitude of the ξ_i 's and a given peak power, the maximum-length sequence gives the largest spectral line energy and seems thus to be a good choice, especially for noisy channels. Both in (ii) and in (iii), λ_0 is different from the other $N - 1$ identical eigenvalues; we will, however, show that β can be selected according to these $(N - 1)$ values and that λ_0 does not affect the convergence if the equalizer is properly preset.

VII. PRESETTING THE TAPS

Since the reference sequence is not synchronized with the received signal, the resulting tap vector may have its main tap in any position,

and it would obviously not make sense to preset in the traditional way of having $c_i = \delta_{i0}$. Instead, we choose an initial tap vector $\mathbf{s}\mathbf{u}$ whose coefficients have equal values s . If we assume $N = 2M + 1$ taps, the initial equalizer transfer function is given by

$$H(\omega) = s \sum_{n=-M}^M e^{-j\omega n T} = s \frac{\sin(N\omega T/2)}{\sin(\omega T/2)}, \quad (54)$$

which is a comb filter with period $1/T$ and attenuation poles at $f = k/NT$, i.e., precisely at the frequencies where the spectral lines of the training sequence are located. Only dc information is thus transmitted to the output prior to the first iteration. This is also obvious from the fact that the output $y = \mathbf{s}\mathbf{x}^T\mathbf{u}$ does not depend on the cyclic shift of \mathbf{x} , since it is the sum of all N sequence samples. If an ideal Nyquist pulse is applied to such a system, the initial distortion of the output signal is very large, i.e.,

$$D_{\text{peak}} = D_{\text{MSE}} = N - 1. \quad (55)$$

This is independent of s . If, however, we look at the average mean-square error of the training sequence, we have with the initial setting

$$\begin{aligned} \epsilon_1^2 &= \frac{1}{N} \sum_i (\mathbf{s}\mathbf{x}_i^T\mathbf{u} - \xi_i)^2 \\ &= s^2(\mathbf{x}^T\mathbf{u})^2 - 2 \frac{s}{N} (\mathbf{x}^T\mathbf{u})(\xi^T\mathbf{u}) + \frac{1}{N} (\xi^T\xi). \end{aligned} \quad (56)$$

This can be differentiated with respect to s , and we find that the initial mean-square error is minimized if we choose

$$s_{\text{opt}} = \frac{1}{N} \frac{\xi^T\mathbf{u}}{\mathbf{x}^T\mathbf{u}}. \quad (57)$$

The quotient associated with $1/N$ represents the dc gain of the channel and is usually close to unity. Since the dc gain of the equalizer is equal to the sum of the tap coefficients, (57) means that the initial settings should be chosen to have the same sum as the final settings in \mathbf{c}_0 (remember that \mathbf{c}_0 is the inverse DFT of the correction function).

Some further physical insight is obtained if the mean-square error after m iterations is studied. This mean-square error may be expressed as³⁶

$$\epsilon_{m+1}^2 = \sum_{i=0}^{N-1} q_{i,m+1}, \quad (58)$$

where the i th error component, $q_{i,m+1}$, is given by

$$q_{i,m+1} = \mu_i |\delta_i^T \mathbf{r}_i|^2 (1 - \beta\lambda_i)^m. \quad (59)$$

The initial value of each component is proportional to its corresponding eigenvalue and to the square of

$$\delta_i^T \mathbf{r}_i = s \mathbf{u}^T \mathbf{r}_i - \mathbf{c}_o^T \mathbf{r}_i, \quad (60)$$

where

$$\mathbf{u}^T \mathbf{r}_i = \begin{cases} 0 & \text{if } i \neq 0 \\ \sqrt{N} & \text{if } i = 0. \end{cases} \quad (61)$$

The values of $\delta_i^T \mathbf{r}_i$ are then obtained as

$$\delta_i^T \mathbf{r}_i = -\mathbf{c}_o^T \mathbf{r}_i = \{-DFT(\mathbf{c}_o)\} \quad i \neq 0 \quad (62)$$

$$\delta_i^T \mathbf{r}_o = N^{-1}(Ns - \mathbf{c}_o^T \mathbf{u}) \quad \text{if } i = 0. \quad (63)$$

Because of what we have said earlier, we see that these coefficients are proportional to the values of the correction function at the line frequencies, except in the case of $i = 0$, where $\delta_i^T \mathbf{r}_o$ is only proportional to the misadjustment at dc. If we select s according to (57), the error component associated with λ_o becomes zero. The constant β is then selected in accordance with the remaining eigenvalues to provide fast convergence.

A few comments are in order for the case of $\mu_o = 0$. This will occur whenever the sequence is dc-free. An example of this property is a maximum-length sequence that is complemented by one additional bit to provide an equal number of ones and zeros (N would then be even). From (59) we see that the error term associated with μ_o is zero; since there is no spectral line at dc, the gain at $\omega = 0$ is obviously immaterial as long as we transmit the training sequence, and convergence and mean-square error are independent of the choice of s . To see how this affects the solution \mathbf{c}_o , we write the relation $A\mathbf{c}_o = \mathbf{v}$ in the form

$$DD^T W^T \mathbf{c}_o = N W^T \mathbf{v}. \quad (64)$$

Assume k eigenvalues in the diagonal matrix DD^T are zero.* Therefore, we have only $N - k$ linear independent equations for the N components of \mathbf{c}_o . The set of solutions for \mathbf{c}_o can be expressed with k independent linear parameters. In the most important case where only $\mu_o = 0$, this ambiguity can be avoided easily by constraining the sum of the tap values. This sum remains constant throughout the equalization process. We can best show that if we look at the sum of the gradient components,

$$\sum_i \mathbf{u}^T \mathbf{x}_i (\mathbf{x}_i^T \mathbf{c} - \xi_i),$$

* This can happen if the test sequence has zero power at some frequencies $f_k = k/NT$ within the transmission band.

which is obviously zero since $\mathbf{u}^T \mathbf{x}_i$ is zero by definition. We would thus choose s in such a way as to match the desired dc gain (which is no longer immaterial if we transmit random data after the initial training period). This is still in accordance with (57) if the quotient of the right-hand side is replaced by the quotient of the spectral densities at dc when data are random.

VIII. INFLUENCE OF NOISE

So far, we have made the assumption that the received samples are noiseless. We give here a coarse analysis of the effects of noise which will show that its influence is, in fact, quite small and may often be neglected. We assume that the taps are calculated from a single-input signal vector which includes noise; that is, the vector \mathbf{x} in (16) now consists of the received signal values plus noise samples. As the equalizer cannot make any distinction between signal and noise components, a tap vector,

$$\mathbf{c}_{or} = (S + R)^{-1} \boldsymbol{\xi}, \quad (65)$$

will evolve instead of $\mathbf{c}_o = S^{-1} \boldsymbol{\xi}$, where R is a noise matrix defined in accordance with (17). We then write the tap difference vector as

$$\mathbf{c}_o - \mathbf{c}_{or} = S^{-1} R \mathbf{c}_{or} \quad (66)$$

if we combine (21) and (65). If a noiseless test sequence were transmitted over the system, there would be some output error because the vector \mathbf{c}_{or} is different from the optimum \mathbf{c}_o . The resulting mean-square error, averaged over the ensemble of \mathbf{c}_{or} 's, would be

$$\epsilon^2 = E\{(\mathbf{c}_o - \mathbf{c}_{or})^\dagger A (\mathbf{c}_o - \mathbf{c}_{or})\}, \quad (67)$$

and can be written as

$$\epsilon^2 = E\{\mathbf{c}_{or}^\dagger R^\dagger (S^{-1})^\dagger A S^{-1} R \mathbf{c}_{or}\}. \quad (68)$$

If we make use of the relation (37), this can be simplified to

$$\epsilon^2 = \frac{1}{N} E\{\mathbf{c}_{or}^\dagger R^\dagger R \mathbf{c}_{or}\}. \quad (69)$$

Assuming that succeeding noise samples are uncorrelated and that $|\mathbf{c}_{or}|^2 \approx 1$, we finally obtain for the mean-square error

$$\epsilon^2 = \sigma^2,$$

where σ^2 is the noise power. We conclude that for reasonable s/n 's there will be only a small bias introduced because of the superimposed noise.

IX. CYCLIC EQUALIZATION USING A MEAN-SQUARE ALGORITHM WITHOUT AVERAGING

The previous discussion has given an analysis of the process of cyclic equalization using the mean-square algorithm with averaging. Much of the insight developed there regarding the final tap values, the type of training sequence to use, etc., carries over to the equalization process which uses the mean-square algorithm without averaging. However, to be more precise we now will carry out an exact analysis of this algorithm. Because it permits a more simple implementation, it is the algorithm without averaging that will most likely be used in practical situations.

Let the N -component tap-signal vector at time $t_o + kT$ be denoted by \mathbf{x}_k . In the absence of noise, succeeding signal vectors will then be related by

$$\mathbf{x}_{k+m} = U^m \mathbf{x}_k, \quad (70)$$

where U is an $N \times N$ cyclic shifting matrix of the form

$$U = \begin{pmatrix} 0 & 0 & \cdots & 0 & 1 \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & & 1 & 0 \end{pmatrix}. \quad (71)$$

Note also that U is orthogonal and

$$U^m = U^{m \pm lN}. \quad (72)$$

The equalizer output at time $t_o + mT$ will be

$$y(t_o + mT) = \mathbf{c}_m^T \mathbf{x}_m = \mathbf{c}_m^T U^m \mathbf{x}_o, \quad (73)$$

where we have expressed the signal vector as a cyclic shift of one fixed state at start-up. We will drop the index on \mathbf{x} from now on.

Let d_k be the reference value of the data signal at $t = t_o + kT$. The mean-square error at $t_o + mT$ is

$$\epsilon_m = E\{(\mathbf{c}_m^T U^m \mathbf{x} - d_m)^2\}, \quad (74)$$

where m indicates any of the equally probable cyclic shifts of \mathbf{x} and d . The expected value in (74) can be obtained by time averaging over $i + 1 \leq k \leq i + N$ because of the cyclic nature of the signals under consideration. The gradient with respect to the tap weights is given by

$$\frac{\partial \epsilon}{\partial \mathbf{c}} = 2E\{e_m U^m \mathbf{x}\}. \quad (75)$$

In this section we make adjustments of \mathbf{c} at each symbol interval and

use a nonaveraged approximation of (75) (the product of error and tap signals of the previous baud interval) for updating. Thus, our strategy becomes*

$$\begin{aligned} \mathbf{c}_{m+1} &= \mathbf{c}_m - \beta U^m \mathbf{x} (\mathbf{c}_m^T U^m \mathbf{x} - d_m) \\ &= U^m (I - \beta \mathbf{x} \mathbf{x}^T) U^{-m} \mathbf{c}_m + \beta U^m \mathbf{x} d_m. \end{aligned} \quad (76)$$

For convenience, we define a data vector ξ which contains the reference values d_m . We further define an N -dimensional vector,

$$\mathbf{r} = \{r_i\}, \quad r_i = \delta_{ik}, \quad (77)$$

containing zeros in all positions except in one reference (k th) position ("center tap"). We observe that

$$d_m = \mathbf{r}^T U^m \xi, \quad (78)$$

and we can write (76) in the form

$$\mathbf{c}_{m+1} = U^m Z U^{-m} \mathbf{c}_m + \beta U^m E U^{-m} \mathbf{r}, \quad (79)$$

where we have introduced for convenience

$$Z = I - \beta \mathbf{x} \mathbf{x}^T \quad (80)$$

$$E = \mathbf{x} \xi^T. \quad (81)$$

By solving the time-varying difference equation (79), the tap vector after m adjustments can be expressed as

$$\mathbf{c}_{m+1} = U^m \left\{ Q^m \mathbf{c}_1 + \beta \sum_{k=0}^{m-1} Q^k E U^{k-m} \mathbf{r} \right\}. \quad (82)$$

The new matrix Q in (82) is defined as

$$Q = Z U^{-1} = (I - \beta \mathbf{x} \mathbf{x}^T) U^{-1}, \quad (83)$$

and will play an important role in further analysis. We can also easily verify the synchronization-invariant properties of (82). In fact, if we replace \mathbf{x} by $U^i \mathbf{x}$ (introducing some arbitrary delay), we obtain

$$\mathbf{c}_{m+1} = U^{m+i} \left\{ Q^m U^{-i} \mathbf{c}_1 + \beta \sum_{k=0}^{m-1} Q^k E U^{k-m} \mathbf{r} \right\}, \quad (84)$$

but since we choose the initial \mathbf{c}_1 with equal bias values for all coefficients, $U^{-i} \mathbf{c}_1 = \mathbf{c}_1$, and (84) and (82) are identical except for an i -position cyclic shift of the resulting tap vector.

* We are assuming β is constant; in practice, it might be desirable to make β dependent on m .

X. SOLVING THE DIFFERENCE EQUATION

Before we discuss (82) in more detail, we define

$$H_m = \beta \sum_{k=0}^{m-1} Q^k E U^k \quad (85)$$

because sums of this type will be frequently needed in the subsequent analysis. Examples are

$$\begin{aligned} H_0 &= 0 \\ H_1 &= \beta E \\ H_N &= \beta \sum_{k=0}^{N-1} Q^k E U^k. \end{aligned}$$

Further, it is rather straightforward to show that

$$H_{i+j} = H_i + Q^i H_j U^i = H_j + Q^j H_i U^j, \quad (86)$$

and, as a special case,

$$H_{iN+n} = H_{iN} + Q^{iN} H_n = H_n + Q^n H_{iN} U^n. \quad (87)$$

H_{iN} may be expressed in a more convenient form if we introduce a new summation index, $iN + j$,

$$H_{iN} = \beta \sum_{i=0}^{iN-1} Q^{iN} \sum_{j=0}^{N-1} Q^j E U^j. \quad (88)$$

The first series can be summed, and we obtain

$$H_{iN} = (I - Q^{iN})(I - Q^N)^{-1} H_N, \quad (89)$$

where we have made the implicit assumption that $I - Q^N$ is non-singular (we will say more about that in a moment).

We are now ready to study (82), which may be written as

$$\mathbf{c}_{m+1} = U^m \{ Q^m \mathbf{c}_1 + H_m U^{-m} \mathbf{r} \}. \quad (90)$$

By setting $m = iN + n$ and combining the first expression in (87) with (89), we obtain

$$\begin{aligned} \mathbf{c}_{iN+n+1} &= U^n \{ Q^{iN+n} \mathbf{c}_1 + Q^{iN} H_n U^{-n} \mathbf{r} \\ &\quad + (I - Q^{iN})(I - Q^N)^{-1} H_N U^{-n} \mathbf{r} \}. \end{aligned} \quad (91)$$

For any nonzero value of β , \mathbf{c}_m will not converge in the usual sense; however, it will reach a steady-state condition that does not depend upon the initial value of \mathbf{c}_1 . In order that this can occur, we require

$$\lim_{l \rightarrow \infty} Q^{iN} = 0, \quad (92)$$

which means that we require the spectral radius $\rho(Q)$ to be less than unity (all eigenvalues inside the unit circle). This will also guarantee the nonsingularity of $I - Q^N$ and thus the existence of (89). The first (transient) term in (91) will then converge to zero, which means that the steady-state solution is independent of the initial tap settings.* The second term will also converge to zero and the steady-state value of \mathbf{c} is

$$\mathbf{c}_{\infty, n} \triangleq U^n(I - Q^N)^{-1}H_N U^{-n}\mathbf{r}. \quad (93)$$

This solution is periodic in n ; it is trivial to verify that, owing to the cyclic nature of U ,

$$\mathbf{c}_{\infty, n+N} = \mathbf{c}_{\infty, n}. \quad (94)$$

As an important special case we have, if $n = 0$,

$$\mathbf{c}_{\infty} = (I - Q^N)^{-1}H_N\mathbf{r} \triangleq H_{\infty}\mathbf{r}. \quad (95)$$

After $m = lN$ iterations, the tap vector is

$$\mathbf{c}_{lN+1} = Q^{lN}\mathbf{c}_1 + (I - Q^{lN})(I - Q^N)^{-1}H_N\mathbf{r}. \quad (96)$$

By combining (95) and (96), we can express the convergence with the error vector $\mathbf{c}_{lN+1} - \mathbf{c}_{\infty}$,

$$\mathbf{c}_{lN+1} - \mathbf{c}_{\infty} = Q^{lN}(\mathbf{c}_1 - \mathbf{c}_{\infty}), \quad (97)$$

as a function of the initial error vector. This is a particularly simple form, which shows how the convergence is directly dependent on the eigenvalues of Q . The error vector is reduced by a factor Q^N with each cycle of iterations. The eigenvalues of Q are functions of β and of the signal format and channel characteristics. We will study this problem in the next section.

XI. CONDITIONS FOR CONVERGENCE

The eigenvalues λ and eigenvectors \mathbf{z} of Q are determined by

$$Q\mathbf{z} = U^{-1}(I - \beta\mathbf{x}\mathbf{x}^T)\mathbf{z} = \lambda\mathbf{z}. \quad (98)$$

We can calculate $(\lambda\mathbf{z})^\dagger(\lambda\mathbf{z})$ and obtain

$$|\lambda|^2\mathbf{z}^\dagger\mathbf{z} = \mathbf{z}^\dagger\mathbf{z} - 2\beta\mathbf{z}^\dagger\mathbf{x}\mathbf{x}^\dagger\mathbf{z} + \beta^2\mathbf{z}^\dagger(\mathbf{x}\mathbf{x}^\dagger)^2\mathbf{z}. \quad (99)$$

Assuming normalization of the eigenvectors, we then require for stability that

$$|\lambda|^2 = 1 - 2\beta|\mathbf{z}^\dagger\mathbf{x}|^2 + \beta^2(\mathbf{x}^\dagger\mathbf{x})|\mathbf{z}^\dagger\mathbf{x}|^2 < 1. \quad (100)$$

* If only a small number of iterations are used for training, \mathbf{c}_1 should be chosen carefully, since \mathbf{c} will then be a function of the transient term as well.

If we now assume that $\mathbf{z}^T \mathbf{x} \neq 0$, we get the simple condition

$$0 < \beta < \frac{2}{\mathbf{x}^T \mathbf{x}} \quad (101)$$

to ensure convergence of the tap coefficients. Note that the bound (101) depends only on the received signal power, which can be normalized with an automatic gain control.

A completely different situation occurs if \mathbf{x} is orthogonal to an eigenvector \mathbf{z} ; $\mathbf{z}^T \mathbf{x} = 0$. It is easy to see from (100) that this would imply $|\lambda| = 1$, regardless of β . This case must be avoided and needs some special attention.

We first conclude that $\mathbf{z}^T \mathbf{x} = 0$ implies that \mathbf{z} is an eigenvector of both Q and U ; this is evident from (98). The next step is then to determine the eigenvectors \mathbf{y} and eigenvalues μ of the cyclic shifting matrix U . They are defined by the equation

$$U\mathbf{y} = \mu\mathbf{y}. \quad (102)$$

We introduce a unitary matrix W with elements

$$w_{ik} = \frac{1}{\sqrt{N}} \exp\left(-j \frac{2\pi}{N} ik\right), \quad 0 \leq i, k \leq N-1 \quad (103)$$

and observe that

$$\{W^T U W\}_{ik} = \delta_{ik} \exp\left(-j \frac{2\pi i}{N}\right) \quad (104)$$

is diagonal. The eigenvalues of U are thus given by

$$\mu_i = \exp\left(-j \frac{2\pi i}{N}\right) \quad i = 0, 1, \dots, N-1 \quad (105)$$

and the corresponding eigenvectors are

$$\mathbf{y}_i^T = (w_{i0}, w_{i1}, w_{i2}, \dots, w_{i,N-1}). \quad (106)$$

We now define a vector \mathbf{h} with values $\mathbf{y}_i^T \mathbf{x}$,

$$\mathbf{h} = \begin{bmatrix} \mathbf{y}_0^T \mathbf{x} \\ \vdots \\ \mathbf{y}_{N-1}^T \mathbf{x} \end{bmatrix} = W\mathbf{x}. \quad (107)$$

The components of \mathbf{h} are thus simply the components of the discrete Fourier transform of \mathbf{x} , and we can finally write our requirement $\mathbf{z}^T \mathbf{x} \neq 0$ in the form

$$\sqrt{N}h_i = \sum_{k=0}^{N-1} x_k \exp\left(-j \frac{2\pi}{N} ik\right) \neq 0 \quad \text{for } i = 0, \dots, N-1. \quad (108)$$

This is not a serious restriction, since most channels will produce an \mathbf{x} whose DFT will have only nonzero components. Difficulties can arise, however, with frequency gaps of severe attenuation within the pass-band range, but this is a condition that needs special attention with any equalizer. Partial-response signaling does not satisfy (108), and we conclude that it cannot be used for cyclic equalization without changes in the equalizer structure or tap-updating algorithm.

Before we leave the stability discussion, we would like to point out another aspect of our problem. By setting $n = 1$ and $lN \rightarrow \infty$ in (89), we obtain

$$H_\infty - QH_\infty U = \beta E,$$

or, after postmultiplying with U^{-1} ,

$$QH_\infty - H_\infty U^{-1} = -EU^{-1}. \quad (109)$$

Matrix equations of the above type play an important role in stability and control theory (Lyapunov), and it is known that a unique solution of (109) exists only if Q and U^{-1} have no common eigenvalues. This would obviously also lead to our conditions (101) and (108).

XII. ASYMPTOTIC BEHAVIOR

The coefficient vector that minimizes the mean-square error of the received sequence is given by

$$\mathbf{c}_o = A^{-1}\mathbf{v}, \quad (110)$$

where A and \mathbf{v} are the signal-correlation matrix and the cross-correlation vector between this sequence and the reference. Our current strategy does not use the gradient (75) in a steepest descent algorithm, nor do we assume that β decreases as the iterations proceed. Thus, it is to be expected that we obtain settings that are biased with respect to (110). We first write (95) in the form

$$(I - Q^N)\mathbf{c}_\infty = H_N \mathbf{r}. \quad (111)$$

From (70) and (83) we can express Q^N as

$$\begin{aligned} Q^N &= (I - \beta \mathbf{x}_N \mathbf{x}_N^T) \cdots (I - \beta \mathbf{x}_2 \mathbf{x}_2^T) (I - \beta \mathbf{x}_1 \mathbf{x}_1^T) \\ &= I - \beta \sum_i \mathbf{x}_i \mathbf{x}_i^T + \beta^2 \sum_{i>k} \mathbf{x}_i \mathbf{x}_i^T \mathbf{x}_k \mathbf{x}_k^T - \cdots (-1)^N \beta^N \mathbf{x}_N \cdots \mathbf{x}_1^T. \end{aligned} \quad (112)$$

If the signal matrix,

$$A = E\{\mathbf{x}_i \mathbf{x}_i^T\} = \frac{1}{N} \sum_{i=1}^N \mathbf{x}_i \mathbf{x}_i^T, \quad (113)$$

is introduced, we have

$$I - Q^N = \beta N A \left\{ I - \frac{\beta}{N} A^{-1} \sum_{i>k} \mathbf{x}_i \mathbf{x}_i^T \mathbf{x}_k \mathbf{x}_k^T + \cdots \right\}. \quad (114)$$

We can expand the right-hand side of (111) in a similar way,

$$\begin{aligned} H_N \mathbf{r} &= \beta \sum_{i=1}^N \left\{ \prod_{j=i+1}^N (I - \beta \mathbf{x}_j \mathbf{x}_j^T) \right\} \mathbf{x}_i d_i \\ &= \beta N \mathbf{v} - \beta^2 \sum_{i=1}^{N-1} \sum_{j=i+1}^N \mathbf{x}_j \mathbf{x}_j^T \mathbf{x}_i d_i + \dots, \end{aligned} \quad (115)$$

where the signal correlation vector \mathbf{v} is defined as

$$\mathbf{v} = E\{\mathbf{x}_i d_i\} = \frac{1}{N} \sum_{i=1}^N \mathbf{x}_i d_i. \quad (116)$$

Combining (113) to (116) and writing out only the first-order terms in β/N yield

$$\begin{aligned} \mathbf{c}_\infty &= \left\{ I + \frac{\beta}{N} A^{-1} \sum_{i>k} \mathbf{x}_i \mathbf{x}_i^T \mathbf{x}_k \mathbf{x}_k^T - \dots \right\} \mathbf{c}_o \\ &\quad - \frac{\beta}{N} A^{-1} \sum_{i>k} \mathbf{x}_i \mathbf{x}_i^T \mathbf{x}_k d_k - \dots. \end{aligned} \quad (117)$$

The neglected terms in (117) are multiplied with higher powers of β . It is, therefore, always possible to choose β small enough to make the linear term dominant. We can conclude that the resulting asymptotic tap vector differs from the MMSE solution \mathbf{c}_{opt} by a bias which, for sufficiently small β , is directly proportional to β and may be made arbitrarily small. Very fast initial convergence can be obtained by choosing β large; then β may be made smaller for the remaining iterations to reduce the bias error (gear shifting).^{*} This will also reduce the periodic fluctuation of the tap coefficients in the final steady state.

On the other hand, one should always keep in mind that the cyclic process is used only during the training time and that random data are used later on for adaptation. The tap vector that yields MMSE for the training sequence generally does not minimize the mean-square distortion for random data. However, the work of Chang and Ho⁴⁰ indicates that (for small β) the results may not be significantly in error. It would be expected that, in the normal data set application, cyclic equalization would be used for enough cycles to achieve a good open eye; then a longer training sequence would be used, decision-directed, to determine the steady-state tap coefficients.

XIII. ACCELERATED PROCESSING

After these theoretical studies, we conclude our paper by discussing some more practical aspects of the signal-processing organization.

^{*} It seems possible that a continuous decrease of β during the iterations would yield superior results; we have, however, not analyzed this case.

More precisely, we present a somewhat modified implementation technique of cyclic equalization that will allow a further reduction in the initial training time. For this, we assume that the received sequence is not substantially corrupted by noise. In a highly dispersive channel with a relatively high s/n , this is a realistic assumption since inter-symbol interference is completely dominant and noise is of minor influence at the beginning of equalization. Once the initial transients have settled, the receiver will thus see a train of continuously repeated identical sequences. No information is lost if one sequence length is stored in the receiver for further processing and the input is switched off. Such a system is depicted in Fig. 10.

When the received sequence is not corrupted by noise, the receiver will see a train of continuously repeated identical sequences. No information is lost if one sequence length is stored in the receiver for further processing and the input is switched off. Such a system is depicted in Fig. 10.

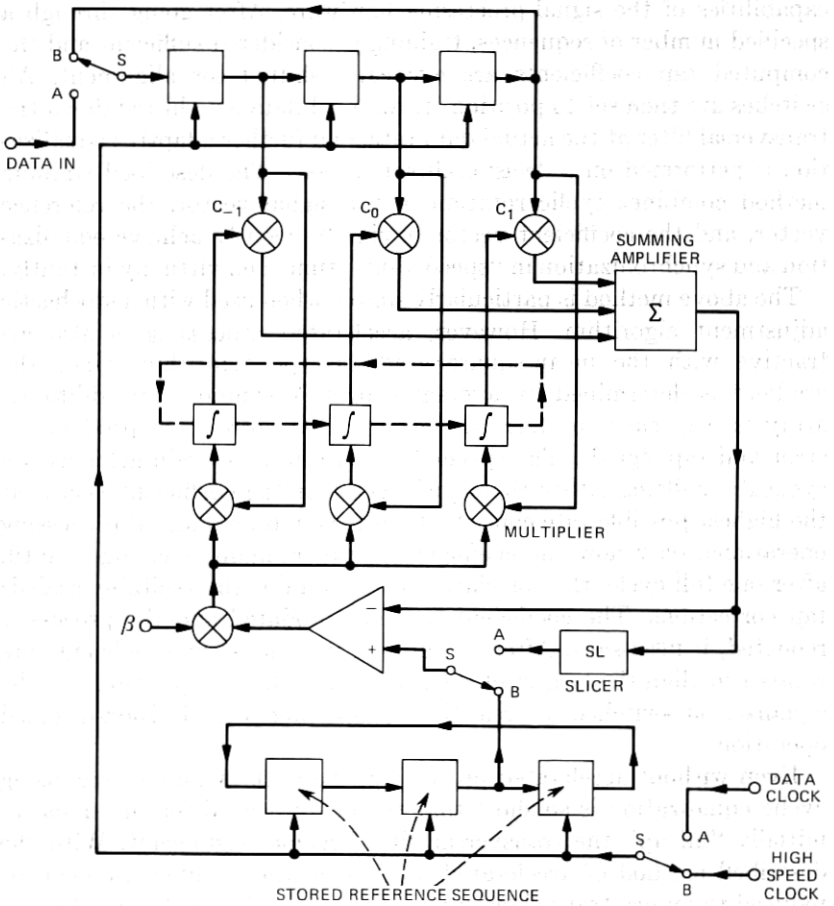


Fig. 10—Block diagram of cyclic equalizer equipped for accelerated processing.

For clarity, only three taps are shown. The samples of the data sequence are entered into the delay line (shift register in the case of a digital equalizer) of the transversal filter while switches S are in position A . As soon as one full sequence is stored, i.e., when samples have reached the end of the delay line, switches S are moved to position B . Thus, a shift-register ring circuit is formed and the stored samples can be shifted cyclically by applying appropriate clock pulses. The stored reference sequence is shifted at the same speed. It is important to realize that this speed need not be related to the actual data rate. The stored signal vector and the reference sequence can be shifted at a much higher rate, thus simulating a "speeded-up" data flow. Initial training can be achieved in a time interval limited only by the speed capabilities of the signal-processing hardware. After going through a specified number of sequences, training is considered sufficient and the computed tap coefficients are cyclically shifted for alignment. All switches are then set to position A , received data are shifted down the transversal filter at the actual data rate, and further adaptive equalization is performed on a decision-directed basis. The described training method combines cyclic rotation of the signal vector, the reference vector, and the coefficient vector to simultaneously achieve equalization and synchronization in "speeded-up" time, i.e., virtually instantly.

The above method is particularly simple when used with a stochastic adjustment algorithm. However, accelerated processing is also attractive with the mean-square gradient-type algorithm. Since the gradient is determined by averaging over N symbols, an additional array is necessary to store the accumulated correlation products of error and tap signals. The speeded-up data flow is again achieved by cyclically shifting either the signal vector or the coefficient vector at the highest possible rate consistent with the required signal-processing operations, only now the coefficient vector remains unchanged until, after one full cycle, the correlator array contains the (suitably scaled) tap corrections. The coefficients are now updated and the process is repeated, if necessary. After a couple of iterations the coefficients are rotated to align the largest of them with the reference position and the equalizer is switched to real-time processing and decision-directed operation.

Even without accelerated processing, the initial training time using cyclic equalization is so short that the delay needed for the signal to initially "fill up" the transversal filter becomes significant. With the described method of accelerated processing, the training time can be reduced to an arbitrary short interval limited only by the speed capabilities of the circuit elements. The "fill-up" time becomes completely

dominant. In the extreme, cyclic training can be achieved within a single symbol interval (after the equalizer is filled up).

XIV. CONCLUSION AND SUMMARY

Cyclic equalization, as presented in this paper, is a new method for initial equalizer training. Its main features are:

- (i) A special training sequence where the number of symbols equals the number of equalizer taps.
- (ii) Very fast start-up with provision for further speeded-up operation, reducing training time theoretically to less than one symbol interval.
- (iii) Ideal reference operation with no synchronization required. The processes of equalization and synchronization are combined in a unique way.
- (iv) Perfect equalization at a set of equally spaced points in the frequency domain.
- (v) Simple and economical implementation.

Cyclic equalization provides a set of tap coefficients that need to be cyclically rotated after initial training. At this time, a coarse equalization is achieved, the eye pattern is open, and the equalizer can switch to a decision-directed mode to achieve final tap settings using random data.

We have shown that the periodic training sequence can always be exactly equalized, so that all unbiased tap-updating algorithms will converge to the same tap settings, namely the inverse DFT of the sampled channel correction function. The mean-square gradient algorithm was analyzed in detail. The channel correlation matrix eigenvalues that influence the convergence are directly related to the lines of the power spectrum of the received sequence. The problem of initial coefficient presetting was discussed, and we made some comments concerning the choice of the training sequence and the influence of noise.

The cyclic equalization process using the mean-square algorithm without averaging was considered, and the difference equation that describes the coefficient convergence was solved. It was proved that the algorithm converges provided that the discrete Fourier transform of the received signal vector has no zero elements, and that the step size is within certain limits related to the number of taps and the received signal power. Finally, it was shown that the tap coefficients for the algorithm without averaging equal those for the algorithm with averaging except for an error term which goes to zero as the step size approaches zero.

The paper has been concluded by presenting a signal processing technique that achieves "accelerated convergence." This allows coefficient calculation in a time interval limited only by the speed capabilities of the equalizer circuitry.

REFERENCES

1. E. Kettel, "Ein automatischer Optimisator fuer den Abgleich des Impulsentzerrers in einer Datenuebertragung," *Archiv der Elektrischen Uebertragung*, 18, No. 5 (May 1964), pp. 271-278.
2. R. W. Lucky, "Automatic Equalization for Digital Communication," *B.S.T.J.*, 44, No. 4 (April 1965), pp. 547-588.
3. R. W. Lucky, "Techniques for Adaptive Equalization of Digital Communication Systems," *B.S.T.J.*, 45, No. 2 (February 1966), pp. 225-286.
4. H. R. Rudin, "Automatic Equalization Using Transversal Filters," *IEEE Spectrum*, 4, No. 1 (January 1967), pp. 53-59.
5. R. W. Lucky and H. R. Rudin, "An Automatic Equalizer for General Purpose Communication Channels," *B.S.T.J.*, 46, No. 9 (November 1967), pp. 2179-2208.
6. R. W. Lucky, J. Salz, and E. J. Weldon, "Principles of Data Communication," New York: McGraw-Hill, 1968.
7. M. J. DiToro, "Communication in Time-Frequency Spread Media Using Adaptive Equalization," *Proc. IEEE*, 56, No. 10 (October 1968), pp. 1653-1679.
8. A. Gersho, "Adaptive Equalization of Highly Dispersive Channels for Data Transmission," *B.S.T.J.*, 48, No. 1 (January 1969), pp. 55-70.
9. J. G. Proakis and J. H. Miller, "An Adaptive Receiver for Digital Signaling Through Channels with Intersymbol Interference," *IEEE Trans. on Information Theory*, IT-15, No. 4 (July 1969), pp. 484-497.
10. K. Moehrmann, "Implementation of an Adaptive Equalizer for Fast Data Transmission Involving a Minimum of Complexity," *Nachrichtentechnische Zeitschrift*, 23, No. 1 (January 1970), pp. 36-42.
11. D. Hirsch and W. J. Wolf, "A Simple Adaptive Equalizer for Efficient Data Transmission," *IEEE Trans. on Communication Technology*, COM-18, No. 1 (February 1970), pp. 5-12.
12. P. Monsen, "Linear Equalization for Digital Transmission Over Noisy Dispersive Channels," Doctoral Thesis, School of Engineering and Applied Science of Columbia University, June 1970.
13. P. Monsen, "Feedback Equalization for Fading Dispersive Channels," *IEEE Trans. on Information Theory*, IT-17, No. 1 (January 1971), pp. 56-64.
14. A. Lender, "Decision Directed Adaptive Equalization Technique for High Speed Data Transmission," *IEEE Trans. on Communication Technology*, COM-18, No. 5 (October 1970), pp. 625-632.
15. R. W. Chang, "Joint Optimization of Automatic Equalization and Carrier Acquisition for Digital Communication," *B.S.T.J.*, 49, No. 6 (July-August 1970), pp. 1069-1104.
16. R. W. Chang, "Joint Equalization, Carrier Acquisition, and Timing Recovery for Data Communication," Conference Record, International Conference on Communications, San Francisco, 1970.
17. B. Wendland, "Abtastsysteme zur adaptiven und nichtadaptiven Entzerrung von Kanaelen," *Nachrichtentechnische Fachberichte*, 37 (1969), pp. 335-352.
18. B. Wendland, "Zur Entzerrbarkeit von Datenkanaelen," *Archiv der Elektrischen Uebertragung*, 24, No. 6 (June 1970), pp. 295-300.
19. G. K. McAuliffe, D. M. Montley, and R. A. Northrup, "Operation and Performance of ADEM," *Nachrichtentechnische Fachberichte*, 37, 1969, pp. 366-378.
20. P. D. Daniell, "Adaptive Estimation with Mutually Correlated Training Sequences," *IEEE Trans. on System Science and Cybernetics*, SSC-6, No. 1 (January 1970), pp. 12-19.
21. L. D. Davison, "Steady-State Error in Adaptive Mean Square Minimization," *IEEE Trans. on Information Theory*, IT-16, No. 4 (July 1970), pp. 382-385.
22. C. W. Niessen and D. K. Willim, "Adaptive Equalizer for Pulse Transmission," *IEEE Trans. on Communication Technology*, COM-18, No. 4 (August 1970), pp. 377-395.

23. L. D. Davisson, "Convergence Probability Bounds for Stochastic Approximation," *IEEE Trans. on Information Theory*, *IT-16*, No. 6 (November 1970), pp. 680-685.
24. R. T. Boyd and F. C. Monds, "Equalizer for Digital Communication," *Electronics Letters*, 7, No. 2 (January 1971), pp. 58-60.
25. K. Moehrmann, "Einige Verfahren zur adaptiven Einstellung von Entzerrern für die schnelle Dateneübertragung," *Nachrichtentechnische Zeitschrift*, 24, No. 1 (January 1971), pp. 18-24.
26. D. T. Magill, "Optimal Adaptive Estimation of Sampled Stochastic Processes," Technical Report No. 6302-3, December 1963, Stanford University.
27. B. Widrow, "Adaptive Filters I: Fundamentals," Technical Report No. 6764-6, December 1966, Stanford University.
28. B. Widrow, L. Griffiths, P. E. Mantey, and G. Burwell, "Adaptive Antenna Systems," Technical Report No. 6778-3, September 1967, Stanford University.
29. K. D. Senne, "Adaptive Linear Discrete-Time Estimation," Dissertation, Department of Electrical Engineering, Stanford University, June 1968.
30. K. D. Senne, "An Exact Solution to an Adaptive Linear Estimation Problem," Technical Report No. SLR-TR-70-0014, Frank J. Seiler Research Laboratory, U. S. Air Force Systems Command.
31. T. J. Schonfeld and M. Schwartz, "A Rapidly Converging First-Order Training Algorithm for an Adaptive Equalizer," *IEEE Trans. on Information Theory*, *IT-17*, No. 4 (July 1971), pp. 431-439.
32. T. J. Schonfeld and M. Schwartz, "Rapidly Converging Second Order Tracking Algorithms for Adaptive Equalization," *IEEE Trans. on Information Theory*, *IT-17*, No. 5 (September 1971), pp. 572-579.
33. D. R. George, R. R. Bowen, and J. R. Sorey, "An Adaptive Decision Feedback Equalizer," *IEEE Trans. on Communication Technology*, *COM-19*, No. 3 (June 1971), pp. 281-293.
34. P. Monsen, "Linear Estimation in an Unknown Quasi-stationary Environment," *IEEE Trans. on Systems, Man, and Cybernetics*, *SMC-1*, No. 3 (July 1971), pp. 216-222.
35. E. Y. Ho, "Optimum Equalization and the Effect of Timing and Carrier Phase on Synchronous Data Systems," *B.S.T.J.*, 50, No. 5 (May-June 1971), pp. 1671-1689.
36. R. W. Chang, "A New Equalizer Structure for Fast Start-up Digital Communication," *B.S.T.J.*, 50, No. 6 (July-August 1971), pp. 1969-2013.
37. G. Ungerboeck, "Theory on the Speed of Convergence in Adaptive Equalizers for Digital Communication," *IBM Journal of Research and Development*, 16, No. 6 (November 1972), pp. 546-555.
38. G. Salomonsson, "An Equalizer with Feedback Filter," *Ericsson Technics*, 28, No. 2, 1972, pp. 57-101.
39. R. T. Sha and D. T. Tang, "A New Class of Automatic Equalizers," *IBM Journal of Research and Development*, 16, No. 6 (November 1972), pp. 556-566.
40. R. W. Chang and E. Y. Ho, "On Fast Start-up Equalization Using Maximum Length Pseudo-Random Training Sequences," *B.S.T.J.*, 51, No. 9 (November 1972), pp. 2013-2027.
41. R. D. Gitlin, E. Y. Ho, and J. E. Mazo, "Passband Equalization for Differentially Phase-Modulated Data Signals," *B.S.T.J.*, 52, No. 2 (February 1973), pp. 129-238.
42. K. H. Mueller and D. A. Spaulding, "Fast Start-up Systems for Transversal Equalizers," U. S. Patent No. 3715666, February 6, 1973.
43. K. H. Mueller and D. A. Spaulding, "Cyclic Equalization—A New Rapidly-Converging Equalization Technique for Synchronous Data Communication," Conference Records, ICC74 Minneapolis.
44. R. D. Gitlin, and J. E. Mazo, "Comparison of Some Cost Functions for Automatic Equalization," *IEEE Trans. on Communications*, *COM-21*, No. 3 (March 1973), pp. 233-237.
45. M. Karnaugh, "Automatic Equalizers Having Minimum Adjustment Time," *IBM Journal of Research and Development*, 17, No. 2 (March 1973), pp. 176-179.
46. J. W. Mark and P. S. Budihardjo, "Joint Optimization of Receive Filter and Equalizer," *IEEE Trans. on Communications*, *COM-21*, No. 3 (March 1973), pp. 264-266.

47. J. W. Mark, "A Note on the Modified Kalman Filter for Channel Equalization," *Proc. IEEE*, 61, No. 4 (April 1973), pp. 481-482.
48. R. D. Gitlin, J. E. Mazo, and M. G. Taylor, "On the Design of Gradient Algorithms for Digitally Implemented Adaptive Filters," *IEEE Trans. on Circuit Theory*, *CT-20*, No. 2 (March 1973), pp. 125-136.
49. S. U. H. Qureshi and E. E. Newhall, "An Adaptive Receiver for Data Transmission Over Time Dispersive Channels," *IEEE Trans. on Information Theory*, *IT-19*, No. 4 (July 1973), pp. 448-457.
50. J. W. Mark and P. S. Budihardjo, "Performance of Jointly Optimized Prefilter—Equalizer Receivers," *IEEE Trans. on Communications*, *COM-21*, No. 8 (August 1973), pp. 941-945.
51. J. Salz, "Optimum Mean-Square Decision Feedback Equalization," *B.S.T.J.*, 52, No. 8 (October 1973), pp. 1341-1373.
52. B. R. Saltzberg, "Intersymbol Interference Error Bounds with Application of Ideal Bandlimited Signaling," *IEEE Trans. Information Theory*, *IT-14*, No. 4 (July 1968), pp. 563-568.