2 Fractionally Spaced Blind Equalization

In the blind method of equalization, where the desired signal or training sequence is not available, some property of the signal is used for the determination of the instantaneous error *e(n)*. The property measurement block in Figure 2.1 determines how much the output of the adaptive filter (equalizer) deviates from the desired property and calculates the instantaneous error. This instantaneous error is then used for updating the adaptive filter coefficient vector **f**(n).

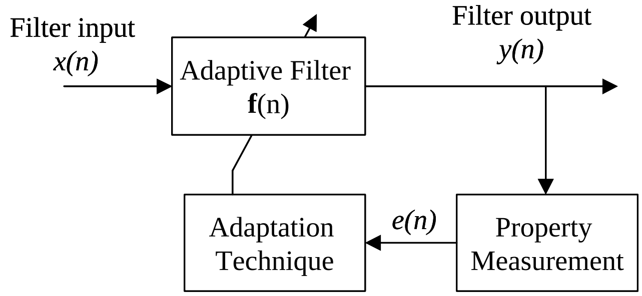

Figure 2.1: Adaptive Filter Model for Blind Equalization.

The most commonly used adaptive algorithm for blind channel equalization is the *Constant Modulus Algorithm* (CMA), which uses the constant modularity of the signal as the desired property. CMA assumes that the input to the channel is a modulated signal that has constant amplitude at every instant in time. Any deviation of the received signal amplitude from the constant value is considered a distortion introduced by the channel. The distortion is mainly caused by band-limiting or multi-path effects in the channel. Both these effects result in inter-symbol interference (ISI) and thus distort the received signal. The objective of equalization is to remove the effect of the channel from the received signal. CMA attempts to accomplish this objective by forcing the output of the adaptive filter (equalizer) to be of constant amplitude. CMA can also be used for QAM signals where the amplitude of the modulated signal is not the same at every instant. The error *e(n)* is then determined by considering the nearest valid amplitude level of the modulated signal as the desired value [1]. Different types of error equations can be used in the *property measurement* block of Figure

2.1 to measure the deviation of the output signal from the desired amplitude level. The next chapter is dedicated to this issue. In the present chapter we will mainly concentrate on the theoretical aspects of different issues arising in blind equalization schemes based mainly on some previous works [1, 2]. In our analysis we will use a *fractionally spaced equalizer* (FSE) as the adaptive filter in Figure 2.1. Therefore, the FSE will first be described with all its necessary features. After that we will discuss some characteristics of CMA, the most commonly used adaptive algorithm in blind equalization schemes.

2.1 Fractionally Spaced Equalizer

2.1.1 Multi-Channel Model

For a fractionally spaced equalizer (FSE), the tap spacing of the equalizer is a fraction of the baud spacing (in time) or the transmitted symbol period. As the output of the equalizer has the same rate as the input symbol rate, the output of the FSE needs to be calculated once in every symbol period. In this situation, the FSE can be modeled as a parallel combination of a number of baud spaced equalizers. This parallel combination of baud spaced equalizers is known as the *Multi-Channel Model* of FSE. The *Multi-Channel Model* of FSE for an over-sampling factor of 2 was derived in one of the earlier works [1]. The over-sampling factor determines the tap spacing of the FSE. If T is the symbol period, then

$$\text{tap spacing} = \frac{T}{oversampling\ \ factor} \tag{2.1}$$

In this section, the general form of the *Multi-Channel Model* of FSE will be derived for an over-sampling factor of M. For the purpose of comparison, the earlier notations [1] will be used in this derivation as well.

For convenience of representation, the combined effect of the linear time-invariant (LTI) channel, and the impulse response of the pulse shaping filter in continuous time will be represented by $c(t)$. Let us consider the continuous time noise as $w(t)$. If $s(n)$ is the transmitted discrete symbol sequence with symbol period T and the base-band equivalent of the analog received signal is $r(t)$, then $r(t)$ can be expressed as:

$$r(t) = \sum_{n=-\infty}^{\infty} s(n)c(t - nT - t_o) + w(t) \tag{2.2}$$

The fractionally spaced equalizer (FSE) is an FIR filter and the tap spacing of this filter is T/M, where M is the over-sampling factor. As the sample period of the input sequence is the same as the tap spacing of the filter, the input sequence to the FSE, $r(t)$, needs to be sampled at intervals T/M apart. The expression for the discrete time equivalent signal $r(k\frac{T}{M})$ is given in (2.3).

$$r(k\frac{T}{M}) = \sum_{n=-\infty}^{\infty} s(n)c(k\frac{T}{M} - nT - t_o) + w(k\frac{T}{M}) \tag{2.3}$$

Here k is an integer and T is the symbol period of the input signal $s(n)$. The time index k will be used to represent the samples in fractional space and the index n will be used to represent the samples in baud space, where the sample rate is the same as the input symbol rate. In fractional space, the output of the equalizer is given by,

$$x(k) = \sum_{i=0}^{MN-1} f(i)r((k-i)\frac{T}{M}) \tag{2.4}$$

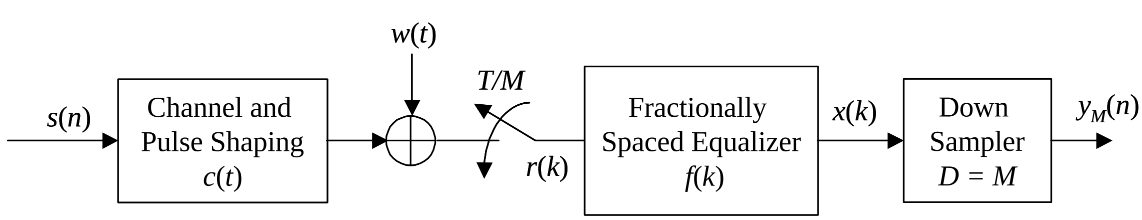

Figure 2.2: Baseband Model for a Communication System with FSE.

Here, the $f(i)$ are the equalizer taps and the tap spacing of the FSE is T/M. The length of the FSE is considered to be MN. As we need only one output symbol corresponding to one input symbol $s(n)$, it is enough to calculate $x(k)$ once every M samples. The other way of making the output rate of the FSE the same as the baud rate is to down sample $x(k)$ by a factor of M. A communication system with a T/M spaced receiver is shown in Figure 2.2.

Now, let us assume that the down sampler is synchronized to collect the *Mth* sample, discarding the first (M-1) samples in a symbol period. The *Mth* sample in every symbol period can be represented as,

$$k = Mn + (M-1), \qquad n = 0,1,2... \tag{2.5}$$

The output of the down sampler, $y_M(n)$ can be expressed as:

$$\begin{aligned} y_M(n) &= \sum_{i=0}^{MN-1} f(i) r((Mn + M - 1 - i)\frac{T}{M}) \\ &= \sum_{i=0}^{MN-1} f(i) r(nT + \frac{(M-1)T}{M} - \frac{iT}{M}) \end{aligned} \tag{2.6}$$

Now (2.6) can be distributed over M terms as follows:

$$\begin{aligned} y_M(n) &= \sum_{i=0}^{N-1}\left[\sum_{j=0}^{M-1} f(Mi+j) r\left(nT + \frac{(M-1)T}{M} - \frac{(Mi+j)T}{M}\right)\right] \\ &= \sum_{i=0}^{N-1}\left[f(Mi) r\left((n-i)T + \frac{(M-1)T}{M}\right) + f(Mi+1) r\left((n-i)T + \frac{(M-2)T}{M}\right)\right. \\ &\quad \left. + \cdots + f(Mi+j) r\left((n-i)T + \frac{(M-j-1)T}{M}\right) + \cdots f(Mi+M-1) r\big((n-i)T\big)\right] \end{aligned} \tag{2.7}$$

Each of the terms in (2.7) represents a convolution. Now, let us define a matrix **F**, which is formed by interleaving the coefficients of the FSE as shown below:

$$\mathbf{F} = \begin{bmatrix} f(0) & f(1) & \cdots & f(M-1) \\ f(M) & f(M+1) & \cdots & f(2M-1) \\ \vdots & \vdots & \ddots & \vdots \\ f((N-1)M) & f((N-1)M+1) & \cdots & f(NM-1) \end{bmatrix} \tag{2.8}$$

Every column of **F** represents a baud spaced filter. Let us define the jth column of **F** as $f_j(n)$ where, $j = 0,1,2,\cdots,M-1$. For each of the $f_j(n)$, $n = 0,1,2,\cdots,N-1$. Similarly, the received sequence $r(k\frac{T}{M})$ can also be sub-divided into M sequences. Each of these sequences will be denoted as $r_j(n)$. Mathematically,

$$r_j(n) = r\left(nT + \frac{M-j}{M}T\right), \qquad \begin{array}{c} j = 0,\cdots M-1 \\ n = 0,1,2,\cdots \end{array} \tag{2.9}$$

Using the definition of $f_j(n)$ and $r_j(n)$, (2.7) can be rewritten as:

$$\begin{aligned} y_M(n) = {} & f_0(n) * r_{M-1}(n) + f_1(n) * r_{M-2}(n) + \cdots \\ & + f_j(n) * r_{M-j-1}(n) + \cdots f_{M-1}(n) * r_0(n) \end{aligned} \tag{2.10}$$

In (2.10), all the '*' indicate the convolution operation. At the end of this section, it will be clear that each of the convolutions in (2.10) corresponds to a sub-channel in the *Multi Channel Model* of FSE. The final stage of deriving the *Multi Channel Model* of FSE is to find the relationship between $s(n)$ and $r_j(n)$. For this purpose (2.9) will be rewritten as:

$$\begin{aligned} r_j(n) &= r\left(nT + \frac{M-j}{M}T\right) \\ &= r\left(\{Mn + (M-j)\}\frac{T}{M}\right) \end{aligned} \tag{2.11}$$

Now using (2.3) and (2.10), $r_j(n)$ can be written in terms of the input symbol sequence $s(n)$ and the channel impulse response as follows:

$$\begin{aligned} r_j(n) &= \sum_{m=-\infty}^{\infty} s(m) c\left(\{Mn + (M-j)\}\frac{T}{M} - mT - t_o\right) + w\left(\{Mn + (M-j)\}\frac{T}{M}\right) \\ &= \sum_{m=-\infty}^{\infty} s(m) c\left((n-m)T + (M-j)\frac{T}{M} - t_o\right) + w\left(nT + (M-j)\frac{T}{M}\right) \end{aligned} \tag{2.12}$$

The fractionally spaced channel $c(kT/M + t_o)$ can be divided into M baud spaced channels by taking one sample out of the M fractionally spaced samples as shown below:

$$c_j(n) = c\left(nT + (M-j)\frac{T}{M} - t_o\right) \tag{2.13}$$

The noise signal can also be subdivided into M groups, namely as follows:

$$w_j(n) = \left(nT + (M - j)\frac{T}{M} \right) \tag{2.14}$$

Using (2.13) and (2.14), (2.12) can be simplified as:

$$r_j(n) = s(n) * c_j(n) + w_j(n) \tag{2.15}$$

Now combining (2.10) and (2.15), the output of the overall system can be written in terms of the input signal sequence, the channel and the equalizer. This is shown below.

$$\begin{aligned} y_M(n) &= \sum_{j=0}^{M-1} f_j(n) * r_{M-j-1}(n) \\ &= \sum_{j=0}^{M-1} f_j(n) * \left(s(n) * c_{M-j-1}(n) + w_{M-j-1}(n) \right) \\ &= s(n) * \left\{ f_0(n) * c_{M-1}(n) + f_1(n) * c_{M-2}(n) + \cdots f_j(n) * c_{M-j-1}(n) + \cdots f_{M-1}(n) * c_0(n) \right\} \\ &\quad + \left\{ f_0(n) * w_{M-1}(n) + f_1(n) * w_{M-2}(n) + \cdots f_j(n) * w_{M-j-1}(n) + \cdots f_{M-1}(n) * w_0(n) \right\} \end{aligned} \tag{2.16}$$

Equation (2.16) leads us to the multi-channel model for fractionally spaced equalization. In this model, the input and output of the system have the same sampling rate. The multi-channel model of the fractionally spaced equalizer, based on (2.16), is shown in Figure 2.3. Each of the channels of the multi-channel model is generally called a *sub-channel* and the equalizer in each branch is called a *sub-equalizer*. It should be noted that the sub-equalizers are not the inverse of their respective sub-channels. The earlier exposition [1] can be derived by setting $M = 2$ in the corresponding equations of this section. A similar set of equations and model can be derived in analogous fashion, if the output decimator (shown in Figure 2.2) collects any sample other than the Mth sample.

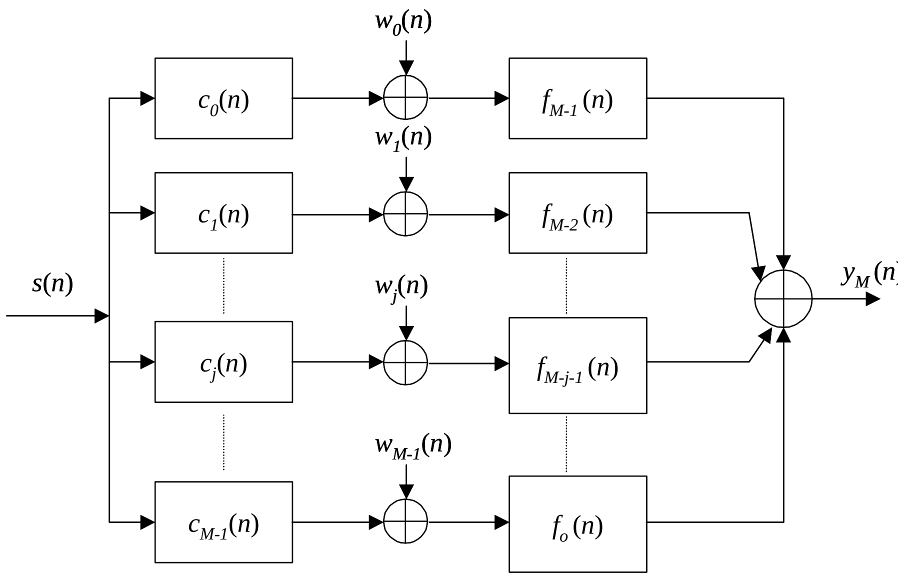

Figure 2.3: Multi-Channel Model of the Fractionally Spaced Equalizer.

2.1.2 Transfer Function and Condition for Perfect Equalization (Constraint on the Channel)

In the noiseless condition the $w_j(n)$ terms in (2.16) will be zero. In that case the impulse response $h(n)$, from the transmitted source to the baud spaced equalizer output, will be the same as the output of the equalizer if $s(n)$ is considered as the Kronecker delta $\delta(t)$. Therefore,

$$h(n) = f_0(n) * c_{M-1}(n) + f_1(n) * c_{M-2}(n) + \cdots f_j(n) * c_{M-j-1}(n) + \cdots f_{M-1}(n) * c_0(n). \quad (2.17)$$

Now the transfer function $H(z)$ of the overall system can be derived by taking the z-transform on both sides of (2.17).

$$H(z) = F_0(z)C_{M-1}(z) + F_1(z)C_{M-2}(z) + \cdots F_j(z)C_{M-j-1}(z) + \cdots F_{M-1}(z)C_0(z) \quad (2.18)$$

$$\begin{aligned} H(z) &= Z\{h(n)\}, \\ C_j(z) &= Z\{c_j(n)\} \\ F_j(z) &= Z\{f_j(n)\} \end{aligned} \quad (2.19)$$

Here $Z\{\bullet\}$ represents the z-transform of the corresponding sequence. For a perfect zero-forcing system, $H(z) = z^{-d}$ (d is a non-negative integer) and (2.18) becomes:

$$z^{-d} = F_0(z)C_{M-1}(z) + \cdots F_j(z)C_{M-j-1}(z) + \cdots F_{M-1}(z)C_0(z) \tag{2.20}$$

Equation (2.20) is known as a Bezout relationship [1]. The Bezout relationship directly leads to the perfect equalization requirement concerning sub-channel roots. Specifically, for the existence of a (finite-length) zero-forcing equalizer, all the sub-channel polynomials, the $C_j(z)$, must not share a common root. For example[1], if the sub-channel polynomials have a common root, a common polynomial, say $G(z) = g_0 + g_1 z^{-1}$, it can be factored out from all the sub-channel polynomials. Let,

$$C_j(z) = G(z)\hat{C}_j(z) \tag{2.21}$$

Then (2.20) can be written as:

$$\begin{aligned} z^{-d} &= (g_0 + g_1 z^{-1})\left\{F_0(z)\hat{C}_{M-1}(z) + \cdots F_j(z)\hat{C}_{M-j-1}(z) + \cdots F_{M-1}(z)\hat{C}_0(z)\right\} \\ &= (g_0 + g_1 z^{-1})\hat{H}(z) \end{aligned} \tag{2.22}$$

Equation (2.22) results in a contradiction, because no finite length polynomial $\hat{H}(z)$, when multiplied by $G(z)$, can result in the delay operator z^{-d}. This is because, in such a case,

$$\hat{H}(z) = \frac{z^{-d}}{(g_0 + g_1 z^{-1})}, \tag{2.23}$$

which represents an (infinite length) IIR system.

Condition: *For perfect equalization with FIR filters, there cannot be a root (factor) common to all the sub-channel polynomials*

[1] The same example is shown in [1] but with T/2-Spaced FSE.

2.1.3 Requirement for Perfect Source Recovery (Constraint on the Equalizer Length)

Let us consider the length of the FSE to be *MN* and the length of the FIR channel (fractional space) to be *ML*. Now, as in (2.8), we can re-write the equalizer coefficients that form the matrix **F**, as shown below:

$$\mathbf{F} = \begin{bmatrix} f(0) & f(1) & \cdots & f(M-1) \\ f(M) & f(M+1) & \cdots & f(2M-1) \\ \vdots & \vdots & \ddots & \vdots \\ f((N-1)M) & f((N-1)M+1) & \cdots & f(NM-1) \end{bmatrix} \tag{2.24}$$

Let, the *j*th column of **F** be defined as:

$$\mathbf{f}_j = \left[f(j) \quad f(M+j) \quad \cdots \quad f((N-1)M+j) \right]^T, \qquad 0 \le j \le M-1 \tag{2.25}$$

For each value of *j*, $\mathbf{f}_j$ is a baud-spaced equalizer of length *N*. Similarly, using the channel coefficients, we can form vectors $\mathbf{c}_j$ ($0 \le j \le M-1$) that correspond to the impulse responses of the baud-spaced sub-channels in the multi-channel model (Figure 2.3).

$$\mathbf{c}_j = \left[c(j) \quad c(M+j) \quad \cdots \quad c((L-1)M+j) \right]^T, \qquad 0 \le j \le M-1 \tag{2.26}$$

The impulse response of the linear system relating $y_M(n)$ and $s(n)$ (refer to Figure 2.3) can be formed by using *M* baud-spaced convolution matrices. The size of the matrices is (*P*´*N*), where *P = L + N – 1*, which is equal to the length of the resultant sequence of the convolution between sequences of length *N* and *L*. Each of the $\mathbf{c}_j$ vectors will be used to construct a convolution matrix named $\mathbf{C}_j$ as shown below.

$$\mathbf{C}_j = \begin{bmatrix} c(j) & 0 & & \vdots \\ c(M+j) & c(j) & & 0 \\ \vdots & c(M+j) & \ddots & c(j) \\ c((L-1)M+j) & \vdots & & c(M+j) \\ 0 & c((L-1)M+j) & \ddots & \vdots \\ \vdots & \vdots & & c((L-1)M+j) \end{bmatrix}_{P \times N} \tag{2.27}$$

The convolution matrices are defined in such a way that the vector resulting from $\mathbf{C}_j \mathbf{f}_{M-j-1}$ is composed of the coefficients from the convolution $c_j(n) * f_{M-j-1}(n)$. Now let us define a matrix **C** and a vector **f**, such that,

$$\begin{aligned} \mathbf{C} &= \begin{bmatrix} \mathbf{C}_1 & \mathbf{C}_2 & \cdots & \mathbf{C}_M \end{bmatrix} \text{ and} \\ \mathbf{f} &= \begin{bmatrix} \mathbf{f}_M^T & \mathbf{f}_{M-1}^T & \cdots & \mathbf{f}_1^T \end{bmatrix}^T. \end{aligned} \tag{2.28}$$

The $P \times MN$ matrix **C** is a Sylvester resultant matrix and **f** is an $MN \times 1$ vector. Using **C** and **f**, the noise-free multi-channel convolution in (2.17) can be written compactly as follows:

$$\mathbf{h} = \mathbf{C}\,\mathbf{f} \tag{2.29}$$

In (2.29) **h** is a length P column vector that corresponds to the coefficients of the baud-spaced impulse response of the overall system in Figure 2.3. Now for *perfect source recovery* (PSR), the output $y_M(n) = s(n-d)$, for some fixed and integer delay d. This condition leads to a *zero forcing* system with an impulse response of

$$\mathbf{h}_d = \begin{bmatrix} 0 & \cdots & 0 & 1 & 0 & \cdots & 0 \end{bmatrix}^T. \tag{2.30}$$

The only non-zero element occurs in the d-th position. Since the size of **h** is P, d must be less than or equal to P.

Now for PSR, the system of equations given by (2.29) must have a solution with $\mathbf{h} = \mathbf{h}_d$ and the solution exists if $\mathbf{h}_d$ lies in the column space of **C** for a particular d. Therefore, the necessary and sufficient condition for *perfect equalization* is that $\mathbf{h}_d$ must lie in the column space of **C**. However, it has been shown [5, 6] that for complete identification of the channel, or in other words, for the existence of the perfect equalizer, the Sylvester matrix **C** must be full row rank. This condition is sometimes referred to as *strong perfect equalization* [1]. When the matrix **C** is not full row-rank [5], the baud-spaced channels $\mathbf{c}_j$ ($1 \le j \le M$) of the multi-channel model share at least one common root. In this situation, as shown in the previous section, the existence of a perfect FIR equalizer is impossible. Therefore, for PSR or for the existence of the perfect FSE, the following conditions must be satisfied:

1. The matrix **C** must have all linearly independent rows; and
2. d must be chosen such that, $\mathbf{h}_d$ lies in the column space of **C**.

To satisfy the first condition, the number of columns of **C** must be greater than or equal to the number of rows. Since, **C** has dimension ($P \times MN$),

$$\begin{aligned} &P = N + L - 1 \leq MN \\ &\Rightarrow (M-1)N \geq L-1 \\ &\Rightarrow N \geq \frac{L-1}{M-1}. \end{aligned} \tag{2.31}$$

If a baud spaced equalizer is used, rather than a FSE, to equalize the effect of an FIR channel (finite L), it is clear from (2.31) that the length of the equalizer will be infinite, since M = 1. Consequently, it is not possible to find a baud-spaced FIR equalizer that can completely equalize the effect of an FIR channel. A similar expression as (2.31) was also derived by Ye Li and Zhi Ding [14].

2.2 Blind Adaptive Algorithm: CMA

Let $s(n)$ be the input to the channel and $x(n)$ be its output, which is the noisy and distorted version of $s(n)$. Let an equalizer be used to remove the channel distortion from the received signal. If **f** is the tap-weight vector of the equalizer of length L, the output of the equalizer $y(n)$ can be represented by

$$\begin{aligned} &y(n) = \mathbf{f}^H \mathbf{x}(n) \\ &\mathbf{x}(n) = \begin{bmatrix} x(n) & x(n-1) & \cdots & x(n-L+1) \end{bmatrix} \end{aligned} \tag{2.32}$$

In (2.32), $\mathbf{x}(n)$ is the equalizer input vector and $\mathbf{f}^H$ represents the Hermitian transpose of **f**.

In the presence of noise, the main objective of designing an equalizer is to minimize the expected value of the squared value of the recovery error,

$$e(n) = s(n-d) - y(n) \tag{2.33}$$

for a particular value of delay d. This criterion provides the best compromise between ISI and noise amplification in a *minimum mean-squared error* (MMSE) sense [1]. Therefore, the tap-weight vector $\mathbf{f}$ of the equalizer is chosen so as to minimize the MSE by using some optimization technique. Generally the optimization techniques are designed to minimize scalar functions that depend on the specific criteria (e.g. MSE). This scalar function is known as the *Cost Function*. A cost function is defined as the transformation from a vector space spanned by the elements of the tap-weight vector to the space of a real scalar. For the MSE criterion, the cost function J_{MSE} is defined as:

$$J_{MSE} = E\left\{|e(n)|^2\right\} = E\left\{ \left|s(n-d) - \mathbf{f}^H \mathbf{x}(n)\right|^2 \right\} \tag{2.34}$$

Even though the MSE criterion provides the optimum (Wiener solution [7]) equalizer, this criterion cannot be used to optimize the equalizer coefficient vector when the source signal $s(n)$ is completely unknown to the receiver. In such cases, at least one of the properties of the input signal $s(n)$ is assumed to be known at the receiver, and the optimization technique used to remove the channel distortion from the received signal is designed to restore the known property to the signal. The most commonly used blind optimization technique is the *Constant Modulus Algorithm* (CMA), which uses constant modularity as the desired property of the output. Generally the error equation used with CMA is defined as:

$$e(n) = |y(n)|^2 - A^2 . \tag{2.35}$$

In (2.35) A is the desired amplitude level. Other kinds of error equations can also be used to improve the performance of CMA. The next chapter is completely dedicated to this error equation issue, and it will therefore not be discussed further in this chapter. CMA optimizes the equalizer coefficients to minimize the cost function J_{CM}, which is defined as:

$$J_{CM} = E\left\{|e(n)|^2\right\} = E\left\{ \left|A^2 - y^2(n)\right|^2 \right\} \tag{2.36}$$

Since (2.36) is not a quadratic (2nd order) equation, like (2.35), the cost surface corresponding to J_{CM} will be a multi-modal surface instead of a simple convex one, which results from J_{MSE}. Therefore, unlike in the MMSE case, CMA always provides multiple solutions for a signal problem and unavoidable ambiguities arise. To reduce the ambiguities or to increase the number of acceptable solutions, the input source can be differentially encoded. If the source symbols are differentially encoded, the absolute phase information is not required during the detection process. The acceptable output sequence for different modulation schemes with differential encoding are shown in Table 2.1. Therefore, an acceptable system impulse response can include a fixed phase shift in addition to the system delay *d*.

Table 2.1: Acceptable Output Sequence for Different Differentially Encoded Modulation Schemes.

Modulation Scheme	**Acceptable Output Sequence: Transmitted Sequence is *s*(*n*)**
BPSK	$y(n) = \pm s(n-d)$
M-ary PSK	$y(n) = e^{j\frac{2p}{M}m} s(n-d), \quad 0 \le \mathrm{m} \le M-1$
16 QAM	$y(n) = e^{j\frac{p}{2}m} s(n-d), \quad 0 \le \mathrm{m} \le 3$

If the MSE criterion is changed to optimize the equalizer coefficients for both the delay *d* and the allowable phase shift r when the source signal is differentially encoded, the amalgamated cost function J_A takes the form shown below [1]:

$$J_A(\mathbf{f}) = \min_{d,\, r} \left\{ \left(\mathbf{Cf} - r\,\mathbf{h}_d\right)^H \left(\mathbf{Cf} - r\,\mathbf{h}_d\right) + \frac{s_w^2}{s_s^2} \mathbf{f}^H \mathbf{f} \right\}. \tag{2.37}$$

In (2.37), s_w^2 is the variance of the noise, s_s^2 is the variance of the source signal, and **C** and **f** are the same as defined in (2.28). Under these conditions the *constant modulus* (CM) criterion closely approximates the MMSE criterion [1]. The results that relate the constant modulus cost function J_{CM} to the MSE cost function J_A can be summarized [1] as below:

1. For a well-behaved channel, i.e., in the absence of common or nearly common sub-channel roots, and if there exists no noise in the system, the minimum points of both J_{CM} and J_A cost surfaces provide the "ideal zero-cost." For this condition the

minimum points of both cost surfaces appear at the same position in the equalizer space even though both cost surfaces do not have the same shape.

2. Both cost surfaces are symmetric with respect to the origin of the **f** plane.

3. In the presence of white noise, both the CM and MSE minima move towards the origin. The J_{CM} and J_A minima move by different amounts, destroying the equivalence of the noiseless case.

4. For common sub-channel roots, even though both surfaces are deformed severely, the global minima of J_{CM} remain in the vicinity of the global minima of J_A.

5. If the channel is under modeled, or if (2.31) is not satisfied, no equalizer setting is capable of achieving zero MSE or CM cost. Moreover, elongating the channel impulse response increases the possibilities for system delay and thus increases the number of J_{A} minima. The number of J_{CM} minima will however not be changed. The positions of the global CM minima remain close to their MSE counterparts.

6. If a gradient descent algorithm is used for optimization, the system delay depends on the initialization of the equalizer coefficients for both cases. Moreover, for both cases, the initialization determines whether the optimization process will converge to a global or to a local minimum point.

The above results justify the robustness of the relationship between J_{CM} and J_A and based on it, a gradient descent algorithm based on the CM criterion, which is known as CMA, is generally used as the adaptive algorithm when the source signal is completely unknown to the receiver.

2.3 Effect of Initialization on a Blind Equalizer

Since the CM surface is unavoidably multi-modal, the choice of initialization affects both the time-to-convergence and the steady-state performance. Depending on the initialization, the optimization process may end up in a global minimum or in a local minimum. To demonstrate this effect of initialization, a simple simulated example will be presented where the effect of the channel will be realized by using a single pole filter. Before going to the example, the weight update equation for CMA needs to be mentioned. Since the derivation of this equation is given in the next chapter, only the required equations will be mentioned here. Since the most commonly used gradient descent algorithm that minimizes the MSE, having knowledge of the source signal, is the LMS (*Least Mean Square*) algorithm, it will be used for comparison purposes. The error equations and the weight update equations for both algorithms are given below:

CMA: Error equation:

$$e(n) = |y(n)|^2 - A^2 \tag{2.38}$$

Weight Update Equation:

$$\mathbf{f}(n+1) = \mathbf{f}(n) - \mu e(n) y^*(n)\mathbf{x}(n) \tag{2.39}$$

LMS [7]**:** Error equation:

$$e(n) = y(n) - s(n-d) \tag{2.40}$$

Weight Update Equation:

$$\mathbf{f}(n+1) = \mathbf{f}(n) - me(n)\mathbf{x}(n) \tag{2.41}$$

In the simulation the system delay ***d*** was considered to be zero. The system diagram used for the simulation is shown in Figure 2.4. Since the channel used in the simulation was a single pole (pole position at z = -0.7) IIR channel, any baud-spaced equalizer of length greater than or equal to 2 is able to equalize the effect of the channel. Therefore, to reduce the complexity of the simulation, instead of using a FSE, a baud spaced equalizer of length two was used in the simulation. Since our main objective is to show the effect of the initialization, 20 different points will be selected on a radius of 2 in **f**-space to initialize the equalizer coefficients. The same source signal sequence of length 4000 will be used for all the different initializations. The adaptation step-size used in the simulation was 0.005. The simulation results for the LMS and CMA algorithms are shown in Figure 2.5 and Figure 2.6 respectively.

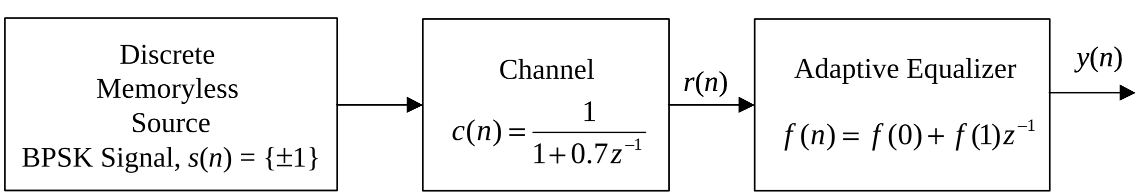

Figure 2.4: A Simple Communication System with Adaptive Equalizer.

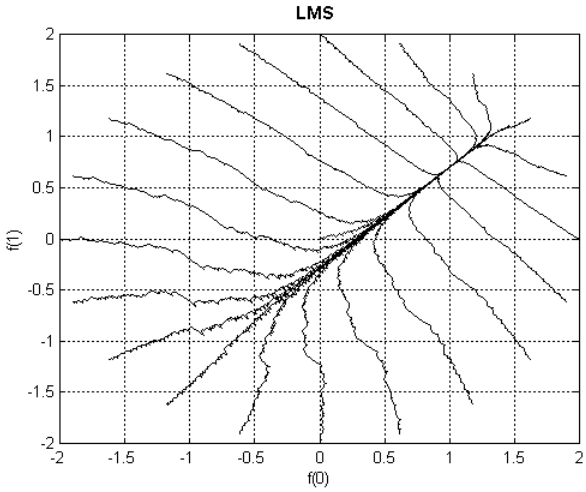

Figure 2.5: Effect of Initialization on LMS Algorithm.

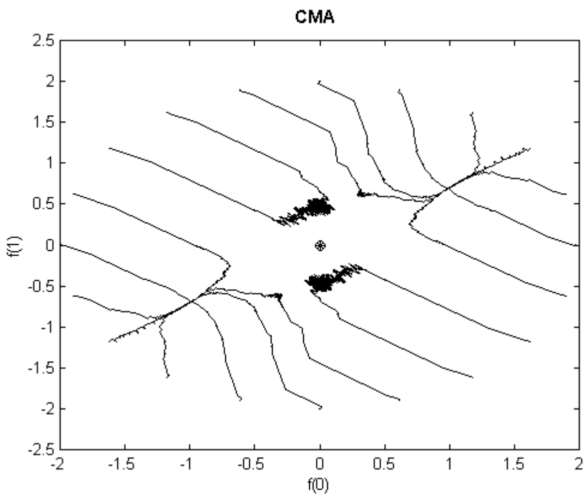

Figure 2.6: Effect of Initialization on CMA Algorithm.

Figure 2.5 shows that if the LMS algorithm is used to adapt the equalizer coefficients, irrespective of the initialization, the algorithm will lead the equalizer coefficients to converge to the true solution which is **f** = [1 0.7]. This is true even if the equalizer is initialized with all zeros. This corroborates that the cost surface of the LMS algorithm has only one minimum point, which is also the global minimum. From Figure 2.6 it is clear that, if CMA is used to update the equalizer coefficients, depending on the initial point in **f**-space, the algorithm leads the equalizer coefficients to converge to four different points. Among these four points, one gives the true inverse of the used channel, which is **f** = [1 0.7]. Now if the differential encoding scheme is used, the negative of the true solution will also be acceptable. Therefore the points corresponding to $\mathbf{f} = \pm[1\ 0.7]$ will be considered to be the global minima. The other two solutions do not give the inverse (or the negative of the inverse) of the channel. Therefore, these two points will be considered to be local minima. In the noiseless case all four minima produce the same cost [1]. The four minima in Figure 2.6 also confirm that the CM criterion provides a multi-modal (multiple minima) cost surface.

The same results were shown earlier, when CMA compared with another type of blind algorithm named a Decision-Directed algorithm [2]. In a Decision-Directed algorithm, a maximum likelihood decision device is used at the output of the equalizer and the error is calculated by subtracting the output of the decision device from the output of the equalizer. The same equation as in LMS is used to update the equalizer coefficients. CMA performs better than the Decision-Directed algorithm by reducing the probability of convergence to local (wrong) minima [2].

Figure 2.6 also shows that it is unwise to initialize the equalizer coefficients with all zeros if CMA is used to update the coefficients. This is because the zero initialized equalizer will result in a zero output and from (2.39) it is evident that for zero output the updating factor will always be zero. Therefore, for this condition the equalizer coefficients and thus the output of the equalizer will always be zero. From this result, it can be concluded that the CM cost surface has always zero gradient at the origin of **f**-space.

2.3.1 Conditions for Convergence to Global Minima

CMA will always converge close to a global minimum [4] if the equalizer coefficients are initialized by using some specific conditions. Before mentioning the conditions, it is necessary to define some of the terms by which these conditions are defined.

Overall System Response:

If $\{c(n)\}$ is the channel impulse response and $\{f(n) : \mathrm{N1} \le n \le \mathrm{N2}\}$ is the equalizer impulse response, the overall system response $\{g(n)\}$ will be defined by the convolution of $\{c(n)\}$ and $\{f(n)\}$as follows:

$$
\begin{aligned}
g(n) &= c(n) * f(n) \\
g(n) &= \sum_{k=N1}^{N2} c(n-k) f(k)
\end{aligned}
\tag{2.42}
$$

Unique Global Minimum Cones:

The unique global minimum cones are defined as:

$$
\begin{aligned}
G_n^+ &= \left\{ \mathbf{g} \in \ell^1(\Re) : g(n) > 0 \quad \text{and } |g(n)| > |g(k)| \quad \text{for all } k \neq n \right\} \\
G_n^- &= \left\{ \mathbf{g} \in \ell^1(\Re) : g(n) < 0 \quad \text{and } |g(n)| > |g(k)| \quad \text{for all } k \neq n \right\}
\end{aligned}
\tag{2.43}
$$

Attainable Set:

The set containing all $\mathbf{g} \in \ell^1(\Re)$, possible for a finite length equalizer, is defined as the *attainable set T*. Therefore,

$$
T = \left\{ \mathbf{g} : g(n) = \sum_{k=N1}^{N2} f(k) c(n-k), \quad f(k) \in \Re \right\}
\tag{2.44}
$$

Kurtosis of the Input Sequence, $s(n)$

The Kurtosis, or the fourth order moment, of the input sequence $s(n)$ is defined as:

$$
\mathrm{K}(s) = E\left\{s^4\right\} - 3E^2\left\{s^2\right\}
\tag{2.45}
$$

Now the conditions that need to be satisfied during the initialization of the equalizer for CMA to converge close to a global minimum are as follows:

1. The initial impulse response of the overall system should be inside a global minimum cone G_n, i.e., $\mathbf{g}^{\mathbf{in}} \in G_n$, where $\mathbf{g}^{\mathbf{in}}$ is the initial overall parameter vector.
2. The initial equalizer setting should satisfy the kurtosis condition, where the Kurtosis condition is defined as:

$$
\frac{kurt(y)}{kurt(s)} > 0.5
\tag{2.46}
$$

In (2.46), y is the output of the equalizer and s is the input to the channel, and *kurt(s)* is defined as:

$$kurt(s) = \frac{\mathrm{K}(s)}{\sigma_s^4} \tag{2.47}$$

Here, s_s^2 is the variance of the sequence $s(n)$.

3. If the position of the global minimum corresponding to G_n is defined by the vector $\mathbf{e}_\mathrm{n}$, $\mathbf{e}_\mathrm{n}$ should be inside or very close to $T \cap G_n$.

Initialization Based on Above Conditions:

For a baud spaced equalizer, condition (1) can be met by initializing f as

$$\begin{aligned} \mathbf{f} &= \left[\, f(-N) \quad \cdots \quad f(-1) \quad f(0) \quad f(1) \quad \cdots \quad f(N) \right]^T \\ &= \left[0 \quad \cdots \quad 0 \quad 1 \quad 0 \quad \cdots \quad 0 \right]^T \end{aligned} \tag{2.48}$$

For this initial value of **f**, the initial overall system impulse response $g^{in}(n) = c(n)$. Usually $\{c(n)\}$ has a unique peak and thus, $\mathbf{g^{in}}$ will be inside a global cone.

For a channel without any zero on the unit circle, condition (3) is satisfied if the equalizer is sufficiently long.

Condition (2) cannot be guaranteed without knowledge of the actual channel. This problem is countered by updating the CMA equalizer in the following way:

a. *Center tap initialization* (described earlier (2.48));
b. *Tap centering*: Center the center of gravity of the tap weight vector for each iteration;
c. *Length extension*: If g has heavy tails after an adequate number of iterations and tap centering, the equalizer length should be extended on the side of the heavy tail (longer equalizer).

Since condition (2) is satisfied indirectly during the update process, instead of during the initialization, the proposed method ([4]) will make the whole system complicated.

Moreover, the authors did not propose the initialization procedure for FSE. Therefore, even though the proposed method of initialization confirms the convergence to the global minima, these methods will not be maintained during the initialization of the equalizer in this work. Instead an easier method will be followed wherein the FSE will be initialized as follows:

$$\mathbf{F} = \begin{bmatrix} f(0) & f(1) & \cdots & f(M-1) \\ f(M) & f(M+1) & \cdots & f(2M-1) \\ \vdots & \vdots & \ddots & \vdots \\ f((N-1)M) & f((N-1)M+1) & \cdots & f(NM-1) \end{bmatrix}$$
$$\mathbf{F}^{in} = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 1 & \cdots & 1 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix} \tag{2.49}$$

The index number of the row with all ones is determined by the overall system delay in baud space. This method of initialization was proposed earlier [1].

2.4 Conventional Blind Equalizer

Let us define a blind equalizer with the following specifications:

- FIR Fractionally Spaced Equalizer
  - Error equation: $e(n) = |y(n)|^2 - A^2$
  - Weight Update Equation: $\mathbf{f}(n+1) = \mathbf{f}(n) - me(n)y^*(n)\mathbf{x}(n)$

Since these are the specifications for most commonly used blind equalizers, any equalizer used in this work having these specifications will be called a *Conventional Blind Equalizer*. This conventional blind equalizer will be used throughout this work for performance comparison with other systems.