3.3. EQUALIZATION

Mobile communication systems require signal processing techniques that improve the link performance in mobile radio environments.

If the modulation bandwidth exceeds the coherence bandwidth of the radio channel, ISI occurs and modulation pulses are spread in time.

Equalization compensates for inter symbol interference (ISI) created by multipath within time dispersive channels.

An equalizer within a receiver compensates for the average range of expected channel amplitude and delay characteristics.

Equalizers must be adaptive since the channel is generally unknown and time varying.

Fundamentals of Equalization

Inter symbol interference(ISI)

- caused by multipath propagation (time dispersion);
- cause bit errors at the receiver;
- the major obstacle to high speed data transmission over mobile radio channels.

Adaptive Equalization and Operating modes of an adaptive equalizer

Training (first stage)

A known fixed-length training sequence is sent by the transmitter so that the receiver's equalizer may average to a proper setting.

The training sequence is designed to permit an equalizer at the receiver to acquire the proper filter coefficients in the worst possible channel conditions

The training sequence is typically a pseudorandom binary signal or a fixed, prescribed bit pattern.

The time span over which an equalizer converges is a function of

- the equalizer algorithm
- the equalizer structure
- the time rate of change of the multipath radio channel.

Equalizers require periodic retraining in order to maintain effective ISI cancellation.

Tracking (second stage)

Immediately following the training sequence, the user data is sent.
As user data are received, the adaptive algorithm of the equalizer tracks the changing channel and adjusts its filter characteristics over time.
commonly used in digital communication systems where user data is segmented into short time blocks.
TDMA wireless systems are particularly well suited for equalizers.TDMA systems send data in fixed-length time blocks, and the training sequence is usually sent at the beginning of a block.
Each time a new data block is received, the equalizer is retrained using the same training sequence .

Communication system with an adaptive equalizer

Equalizer can be implemented at baseband or at IF in a receiver.
Since the baseband complex envelope expression can be used to represent band pass waveforms and, thus, the channel response, demodulated signal, and adaptive equalizer algorithms are usually simulated and implemented at baseband .
Block diagram of a simplified communications system using an adaptive equalizer at the receiver is shown in figure 4.1.1.

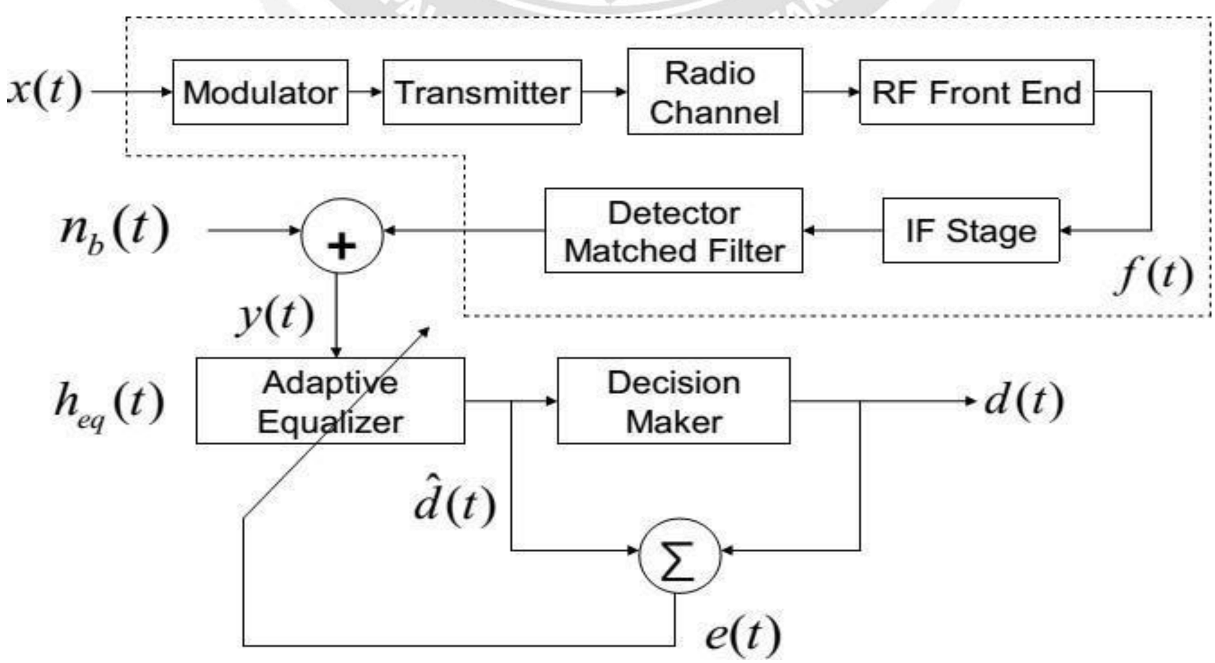

Fig4.1.1:Adaptive Equalizer
[Source : "Wireless communications" by Theodore S. Rappaport,Page-302]

If x (t) is the original information signal, and f(t) is the combined complex baseband impulse response of the transmitter, channel,and the RF/IF sections of the receiver, the signal received by the equalizer may be expressed as

$$y(t) = x(t) \otimes f^*(t) + n_b(t)$$

where f* (t) is the complex conjugate of f(t) , nb (t) is the baseband noise at the input of the equalizer, and 0 denotes the convolution operation. If the impulse response of the equalizer is heq (t), then the output of the equalizer is

$$\begin{aligned}\hat{d}(t) &= x(t) \otimes f^*(t) \otimes h_{eq}(t) + n_b(t) \otimes h_{eq}(t) \\ &= x(t) \otimes g(t) + n_b(t) \otimes h_{eq}(t)\end{aligned}$$

where g (t) is the combined impulse response of the transmitter, channel, RF/IF sections of the receiver, and the equalizer. The complex baseband impulse response of a transversal filter equalizer is given by

$$h_{eq}(t) = \sum_n c_n \delta(t - nT)$$

where Cn are the complex filter coefficients of the equalizer. The desired output of the equalizer is x(t), the original source data. Assume that nb(t) = 0. Then, in order to force d (t) = x (t) in equation(above) , g (t) must be equal to

$$g(t) = f^*(t) \otimes h_{eq}(t) = \delta(t)$$

In the frequency domain, the above equation can be expressed as

$$H_{eq}(f) F^*(-f) = 1$$

where Heq (f) and F(-f) are Fourier transforms of heq (t) and f(t), respectively.

A Generic Adaptive Equalizer

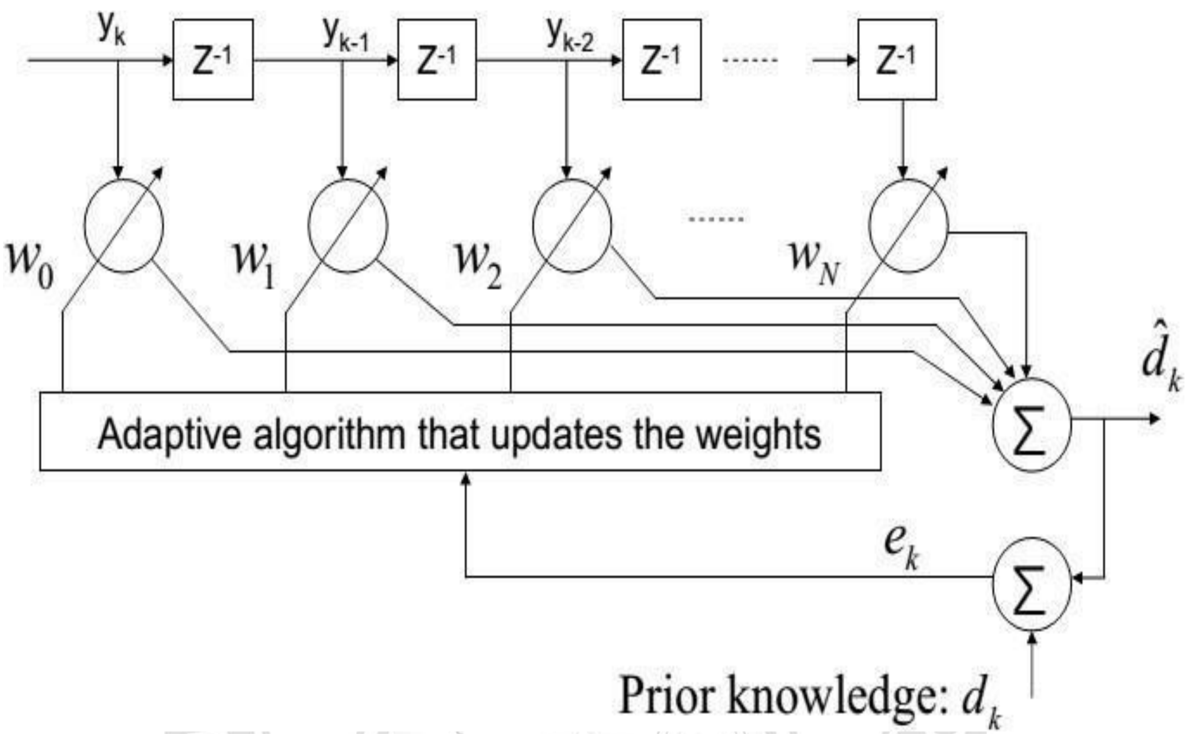

Fig4.1.2: A Generic Equalizer

[Source : "Wireless communications" by Theodore S. Rappaport,Page-303]

The value of Yk depends upon the instantaneous state of the radio channel and the specific value of the noise (as shown in figure 4.1.2) . Yk is a random process.

The adaptive equalizer structure is called a transversal filter, and it has N delay elements, N + I taps, and N + I tunable complex multipliers,called weights.

These weights are updated continuously by the adaptive algorithm either on a sample by sample basis or on a block by block basis.

▪ The adaptive algorithm is controlled by the error signal ek.

ek is derived by comparing the output of the equalizer with some signal which is either an exact scaled replica of the transmitted signal xk or which represents a known property of the transmitted signal.

A cost function is used, the cost function is minimized by using ek, and the weights are updated iteratively.

For example, The least mean squares(LMS) algorithm can serve as a cost function. Iterative operation based on LMS algorithm.

New weights = Previous weights + (constant) x (Previous error) x (Current input vector) , Where

Previous error = Previous desired output — Previous actual output

This process is repeated rapidly in a programming loop while the equalizer attempts to converge Upon reaching convergence, the adaptive algorithm freezes the filter weights until the error signal exceeds an acceptable level or until a new training sequence is sent.

……………………………………..

EQUALIZERS IN A COMMUNICATION RECEIVER

In communication systems the instantaneous combined frequency response will not always be flat, resulting in some finite prediction error.

Because the noise nb (t) is present, an equalizer is unable to achieve perfect performance.

Thus there is always some residual ISI and some small tracking error.

Noise makes equation hard to realize in practice. Therefore, the instantaneous combined frequency response will not always be flat, resulting in some finite prediction error.

Because adaptive equalizers are implemented using digital logic, it is most convenient to represent all time signals in discrete form.

Let T represent some increment of time between successive observations of signal states.

Letting t = tn where n is an integer that represents time = n T, time waveforms may be equivalently expressed as a sequence on n in the discrete domain. Using this notation, equation may be expressed as

$$\hat{d}(n) = x(n) \otimes g(n) + n_b(n) \otimes h_{eq}(n)$$

The prediction error is

$$e(n) = d(n) - \hat{d}(n) = d(n) - [x(n) \otimes g(n) + n_b(n) \otimes h_{eq}(n)]$$

The mean squared error E [|e (n)|^2 is one of the most important measures of how well an equalizer works. E [|e (n)|^2 is the expected value (ensemble average) of the squared prediction error e (n) but time averaging can be used if e (n) is ergodic.

Equalization techniques can be sub divided into two categories:

Linear equalization

The output of the decision maker is not used in the feedback Path to adapt the equalizer.

Nonlinear equalization

The output of the decision maker is used in the feedback path to adapt the equalizer. Many filter structures are used to implement linear and nonlinear equalizers

Classification of equalizers

If the equalizer output is not used in the feedback path to adapt the equalizer, the equalization is linear.

If the equalizer output is fed back to change the subsequent outputs of the equalizer, the equalization is nonlinear.

Linear transversal equalizer (LTE) is made up of tapped delay lines as shown in figure 4.1.3, with the tappings spaced a symbol period (Ts) a part .The transfer function can be written as a function of the delay operator or assuming that the delay elements have unity gain and delay Ts.

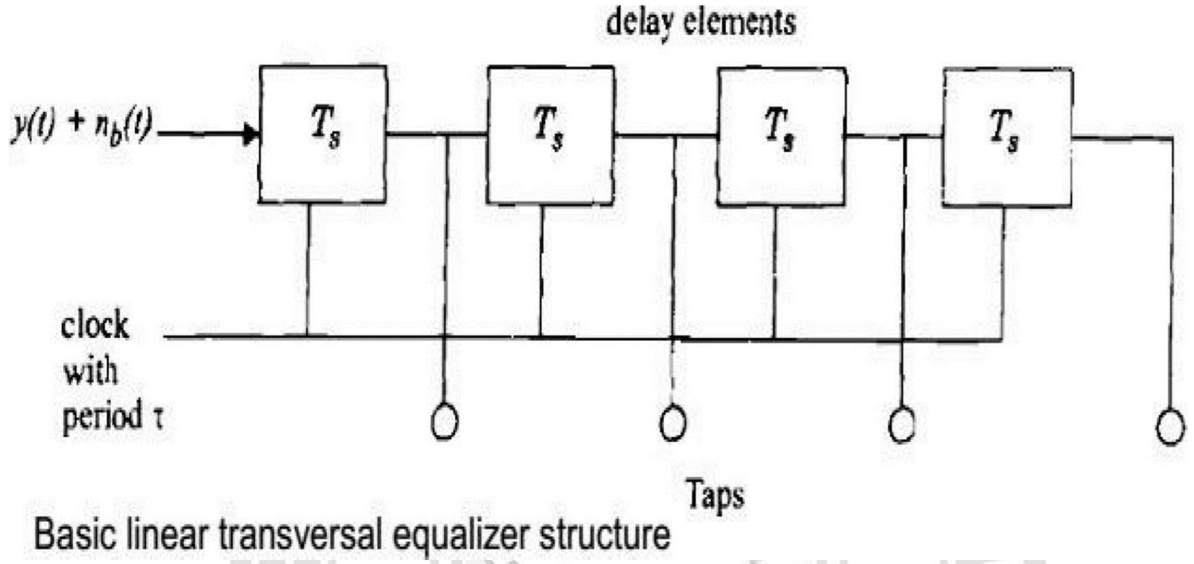

Fig4.1.3: Basic structure of transversal equalizer
[Source : "Wireless communications"by Theodore S. Rappaport,Page-309]

Structure of linear transversal equalizer

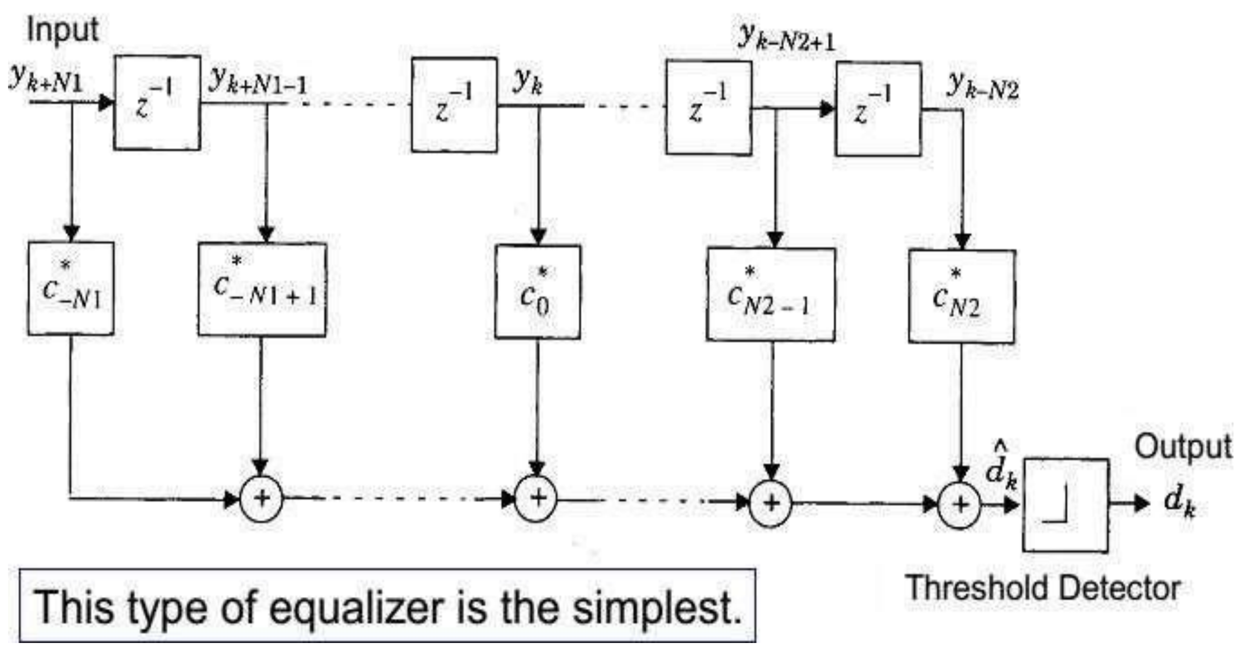

Fig4.1.4:Equalizer
[Source : "Wireless communications" by Theodore S. Rappaport,Page-311]

Linear equalizer is also called Transversal filter as shown in figure 4.1.4.

In linear equalizer, the current and past values of the received signal are linearly weighted by the filter coefficient and summed to produce the output, as shown in figure. If the delays and the tap gains are analog, the continuous output of the equalizer is sampled at the symbol rate and the samples are applied to the decision device.

The implementation is, usually carried out in the digital domain where the samples of the received signal are stored in a shift register.

The output of this filter before a decision is made (threshold decision) is

$$\hat{d}_k = \sum_{n=-N_1}^{N_2} C_n^* y_{k-n}$$

Cn* represents the complex filter coefficients or tap weights,

d**k** is the output at time index k, y**i** is the input received signal at time t**0**+iT,t**0** is the equalizer starting time and $N=N_1+N_2+1$ is the number of taps.

The values N_1 and N_2 denote the number of taps used in the forward and reverse portions of the equalizer.

The minimum mean squared error E [|e (n)|^2] that a linear transversal equalizer can achieve is

$$E[|e(n)|^2] = \frac{T}{2\pi}\int_{-\pi/T}^{\pi/T} \frac{N_0}{|F(e^{j\omega T})|^2 + N_0} d\omega$$

$F(e^{j\omega T})$ is the frequency response of the channel, and No is the noise spectral density.

The linear equalizer can also be implemented as a lattice filter, whose structure is shown in Figure 4.1.5.

The input signal Yk is transformed into a set of N intermediate forward and backward error signals, fn(k) and bn(k) respectively, which are used as inputs to the tap multipliers and are used to calculate the updated coefficients

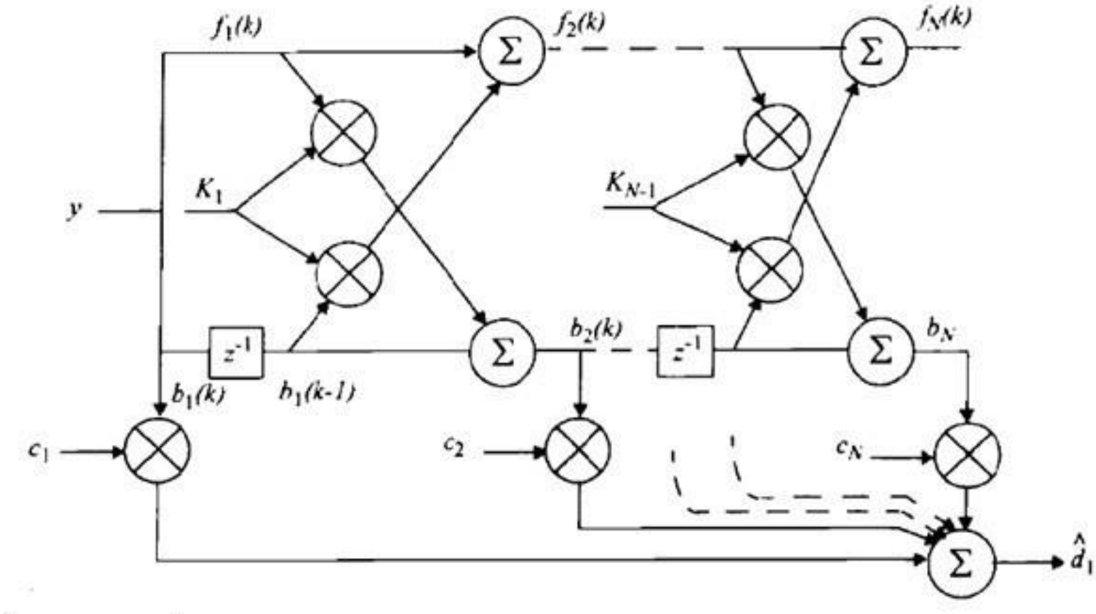

Fig4.1.5: Lattice Equalizer

[Source : "Wireless communications" by Theodore S. Rappaport,Page-312]

Each stage of the lattice is then characterized by the following recursive equations .

$$f_1(k) = b_1(k) = y(k)$$

$$f_n(k) = y(k) - \sum_{i=1}^{n} K_i y(k-i) = f_{n-1}(k) + K_{n-1}(k) b_{n-1}(k-1)$$

$$b_n(k) = y(k-n) - \sum_{i=1}^{n} K_i y(k-n+i)$$

$$= b_{n-1}(k-1) + K_{n-1}(k) f_{n-1}(k)$$

Where Kn(k) is the reflection coefficient for the n th stage of the lattice. The backward error signals, are then used as inputs to the tap weights, and the output of the equalizer is given by

$$\hat{d}_k = \sum_{n=1}^{N} c_n(k) b_n(k)$$

Two main advantages of the lattice equalizer is its numerical stability and faster convergence. The unique structure of the lattice filter allows the dynamic assignment of the most effective length of the lattice equalizer. Hence, if the channel is not very time dispersive, only a fraction of the stages are used. When the channel becomes more time dispersive, the length of the equalizer can be increased by the algorithm without stopping the operation of the equalizer.