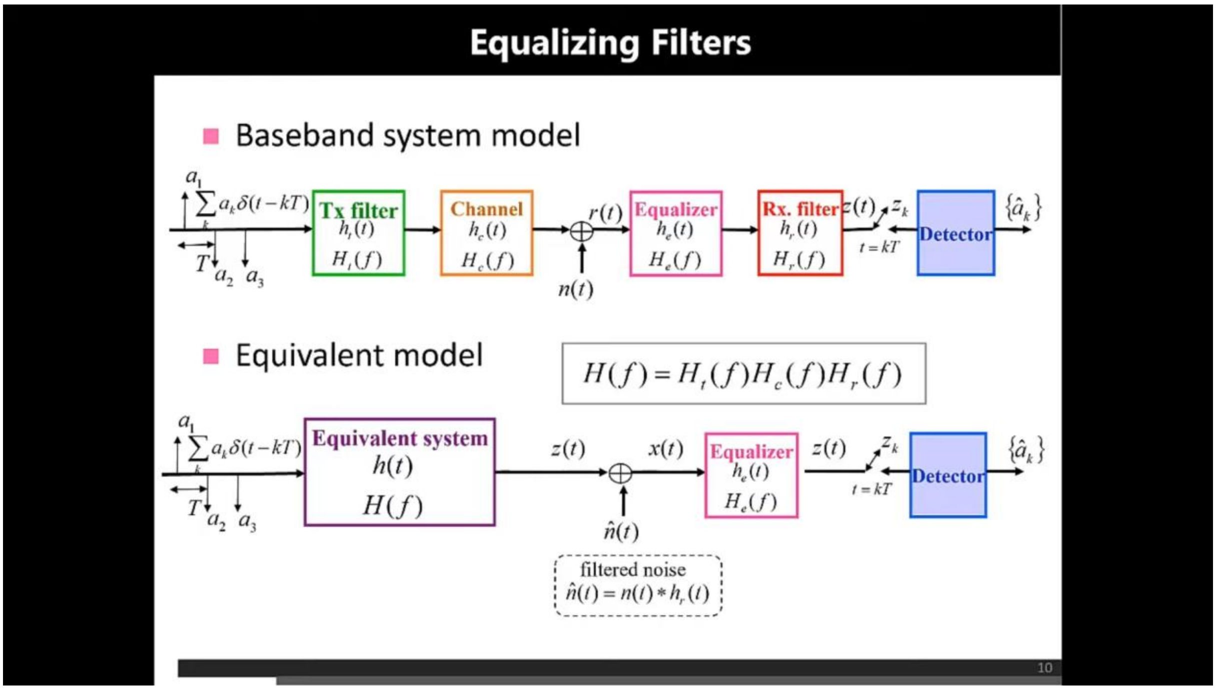
Chapter 3: Radio channel equalization

1- Formulation of the problem



Receiver structure

Typically, in a wireless channel the received symbol $y(k)$ is written as

$$y(k) = h\,x(k) + n(k)$$

Where h is the fading channel coefficient, $x(k)$ is the transmitted symbol and $n(k)$ is the noise sample.

In this model we observe that the current received symbol $y(k)$ depends only of the current transmit symbol $x(k)$.

However, it might frequently happen that the received symbol $y(k)$ is of the form

$$y(k) = h(0)x(k) + h(1)x(k-1) + n(k)$$

Where $x(k-1)$ is the symbol transmitted at time $(k-1)$.

So, the received symbol depends also of the previous transmitted signal. We observe that the symbol $x(k-1)$ interfering with $x(k)$, this phenomenon is termed as intersymbol interference ISI. This ISI leads to performance degradation at the receiver, so removing this ISI is important at the receiver. The process of removing this ISI is called as channel **equalization**.

There are two schems for equalizing (compensating) of the signal distortion, linear equalizers using transvesal filter and non-linear equalizers using the decision feedback filter.

For times variant channel response, the equalizer parameters need to be updated periodically during transmission, this latter is called adaptive equalizer.

3.2. linear equalizers

The linear equalizers are based on transversal filter wich is a weighted tap delyed line that reduce the effect of ISI by proper adjustment of the filter taps, the transversal filter is shown in the figure:

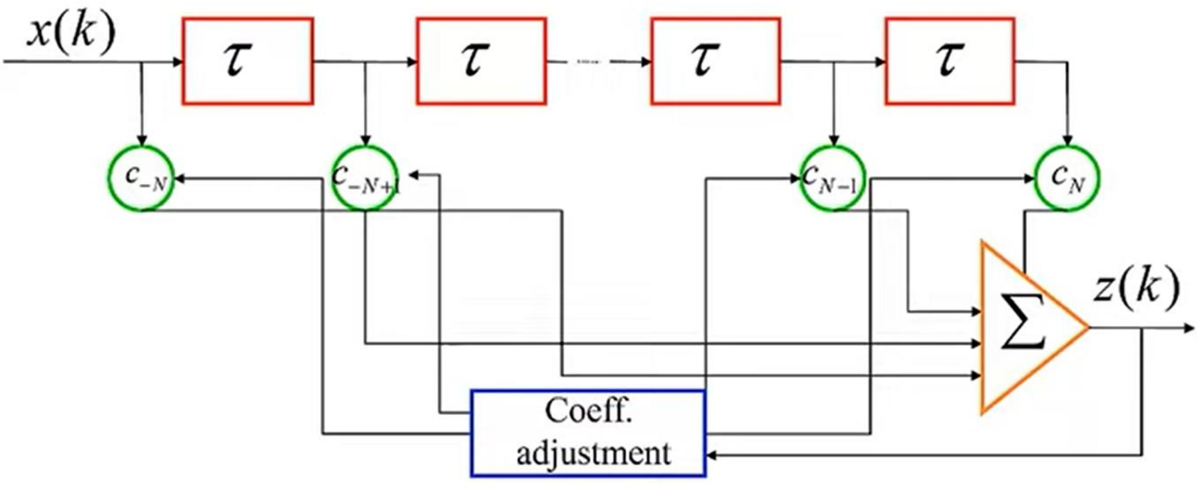


Figure 3.1. Transversal Filter

the received signal is:

$$z(k) = c_n * x(k)$$

$$z(k) = \sum_{n=-N}^{+N} x(k-n)c_n \qquad , \begin{cases} n = -N, \dots, +N \\ k = -2N, \dots, +2N \end{cases}$$

For

$$z(k) = x(k+N)c_{-N} + x(k+N-1)c_{-N+1} + \cdots + x(k-N)c_N$$

$$z(k) = [x(k+N) \;\; x(k+N-1) \;\; \dots \;\; x(k-N)] \begin{bmatrix} c_{-N} \\ \vdots \\ c_0 \\ \vdots \\ c_N \end{bmatrix}$$

$$for\ k = -2N: z(-2N) = [x(-N)\ \ x(-N-1)\ \ \dots\ x(-3N)]\begin{bmatrix} c_{-N} \\ \vdots \\ c_0 \\ \vdots \\ c_N \end{bmatrix}$$

$$for\ k = -2N+1: z(-2N+1) = [x(-N+1)\ \ x(-N)\ \ \dots\ x(-3N+1)]\begin{bmatrix} c_{-N} \\ \vdots \\ c_0 \\ \vdots \\ c_N \end{bmatrix}$$

$$for\ k = 0: z(0) = [x(N)\ \ x(N-1)\ \ \dots\ x(-N)]\begin{bmatrix} c_{-N} \\ \vdots \\ c_0 \\ \vdots \\ c_N \end{bmatrix}$$

$$for\ k = 2N: z(2N) = [x(3N)\ \ x(3N-1)\ \ \dots\ x(N)]\begin{bmatrix} c_{-N} \\ \vdots \\ c_0 \\ \vdots \\ c_N \end{bmatrix}$$

So we can written:

$$Z = \begin{bmatrix} z(-2N) \\ \vdots \\ z(0) \\ \vdots \\ z(2N) \end{bmatrix}, X = \begin{bmatrix} x(-N) & 0 & 0\,0 \dots & 0 & 0 \\ x(-N+1) & x(-N) & 0\ \dots & 0 & 0 \\ \vdots & \vdots & \vdots\ \ \vdots & \vdots & \vdots \\ x(N) & x(N-1) \dots & \dots. & \dots. & x(-N) \\ \vdots & \vdots & \vdots\ \ \vdots & \vdots & \vdots \\ 0 & 0 & 0\ \ 0 & x(N) & x(N-1) \\ 0 & 0 & 0\ \ 0 & 0 & x(N) \end{bmatrix}, C = \begin{bmatrix} c_{-N} \\ \vdots \\ c_0 \\ \vdots \\ c_N \end{bmatrix}$$

$$(4N+1) \times 1 = (4N+1) \times (2N+1) = (2N+1)$$

$$Z = XC$$

3.2.1. Zero Forcing equalizer

The zero forcing solution minimizes the ISI distortion by selecting the c_n weights so that the equalizer output is forced to zero at the *N* sample points either side of the desired pulse. So, the weights are chosen as:

$$z(k) = \begin{cases} 1\ for\ k = 0 \\ 0\ for\ k \neq 0 \end{cases}$$

So the ZF equalizer is:

$$C = X^{-1}Z$$

In order to find C it must that X is square matrix with rows and colomns each having the same dimention as the number of elements of C (2N+1), so we will reduce the number of rows 4N+1 of X to 2N+1 rows by descarting N rows from the top and N rows from the bottom.

3.2.2. Minimum Mean Square Error (MMSE) equalizer

A more robust equalizer is obtained if the tap weitghts c_n are chosen to minimize the mean square error MSE of all ISI termes plus noise power at the output of the equalizer.

We have:

$$Z = XC$$

Multiplaying both sides by X^T

$$X^T Z = X^T XC$$

$$R_{xz} = R_{xx} C$$

Where R_{xz} is the cross-correlation and R_{xx} is the autocorrelation, R_{xz} and R_{xx} are unknown a priori, but can be approximated by transmiting a test signal over the channel. The taps weights are obtained by solving the equation:

$$C = R_{xx}^{-1} R_{xz}$$

4.Adaptive linear equalizers

If the channel characteristics change with time, the taps weights need to be updated.

$$Z = XC$$

In this scheme, an arbitrary weights is obtaind as:

$$C^{k+1} = C^k + \Delta . e^k . X$$

$e^k$is the error signal between the training sequence and the equalizer output Z^k

$$e^k = a^k - Z^k$$

$\Delta$is the step size parameter that affects the convergence of the iteration method

$$\Delta = \frac{0.2}{(2\, N_{taps} + 1) P_r}$$

N_{taps} : nomber of taps

P_r: power of the received signal