3F4: Data Transmission

Handout 7: Channel Equalisation

Ramji Venkataramanan

Signal Processing and Communications Lab, CUED

ramji.v@eng.cam.ac.uk

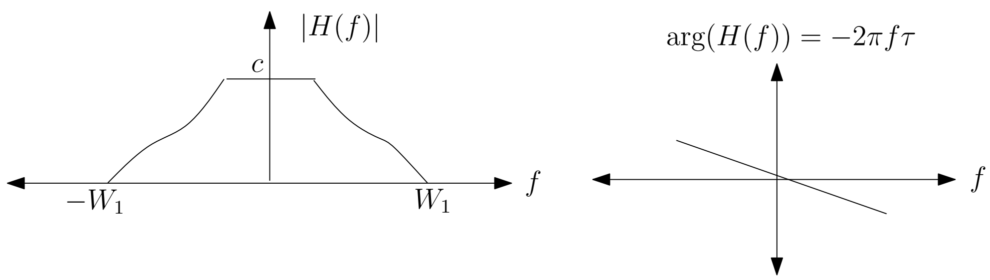
Lent Term 2019

So far, we considered baseband/passband channels whose frequency response $H(f)$ was flat throughout the transmission frequency band.

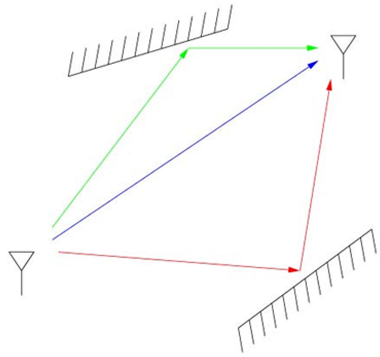
This meant that (after compensating for attenuation and delay), the effective channel was an AWGN channel.

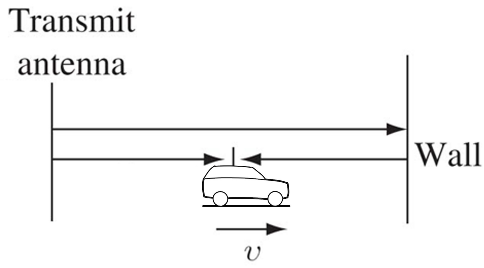
However in many applications such as DSL and mobile communication, the channel spectrum is not flat throughout the transmission band.

For example, the frequency response of a DSL cable might look like:



Also recall from Handout 2 that the mobile wireless channel is a *multi-path channel*.





From "Fundamentals of Wireless Communication", Tse and Viswanath, CUP 2005

For example, if there are L signal paths with delays $\tau_1, \ldots, \tau_L$ and attenuations $\alpha_1, \ldots, \alpha_L$, the received signal is

$$\begin{aligned} y(t) &= \alpha_1\, x(t-\tau_1) + \alpha_2\, x(t-\tau_2) + \ldots + \alpha_L\, x(t-\tau_L) + n(t) \\ &= x(t) \star h(t) \;+\; n(t), \end{aligned}$$

where $h(t) = \alpha_1 \delta(t-t_1) \;+\; \alpha_2 \delta(t-t_2) \;+ \ldots +\; \alpha_L \delta(t-t_L)$.

In general

$$y(t) = \int h(u) x(t-u) du + n(t)$$

A channel frequency response $H(f)$ that is not flat in the transmission band is called *frequency-selective* or *dispersive*

Let us examine the effect of a frequency selective channel response on the demodulation of PAM/QAM.

- Consider the baseband signal $x(t) = \sum_k X_k p(t-kT)$.
- The received signal is

$$y(t) = \int h(u) x(t-u) du + n(t) = \sum_k X_k f(t-kT) \;+\; n(t),$$

where $f(t) = p(t) \star h(t)$

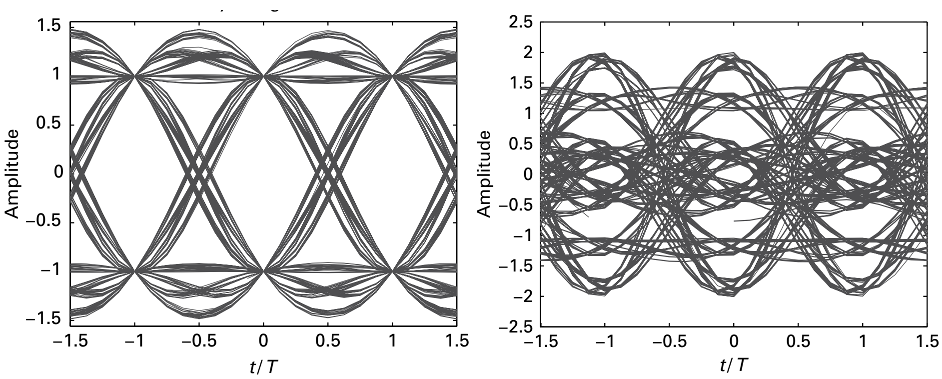
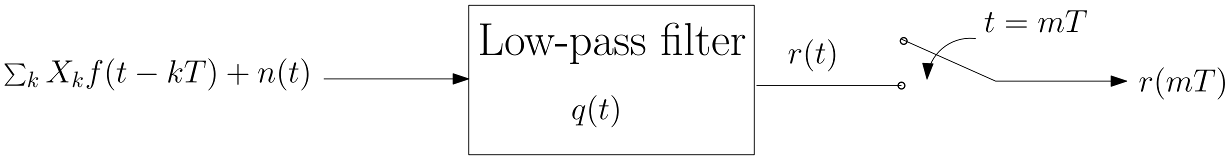


Figure from *Fundamentals of Digital Communication* by U. Madhow, CUP 2008

The right panel shows the eye diagram of BPSK with raised cosine $p(t)$ passed through a channel $h(t) = \delta(t) - 0.6\delta(t-0.5T) + 0.7\delta(t-1.5T)$

At the receiver:



In the absence of noise, note that $r(t) = \sum_k X_k g(t - kT)$, where the overall filter is

$$g(t) = f(t) \star q(t) = p(t) \star h(t) \star q(t).$$

Recall that there will be no ISI (i.e., $r(mT) = X_m$) if that overall filter $g(t)$ satisfies Nyquist pulse criterion.

- But $h(t)$ is determined by the channel $\Rightarrow$ we have to design transmit and receive filters $p(t), q(t)$ tailored to each channel.
- This is not desirable as we want to have standard transmit/receive filters (e.g. root raised cosine), that can be used for many channels.
- Moreover, the channel frequency response typically changes with time, especially in wireless communication.

So we need another technique to deal with ISI.

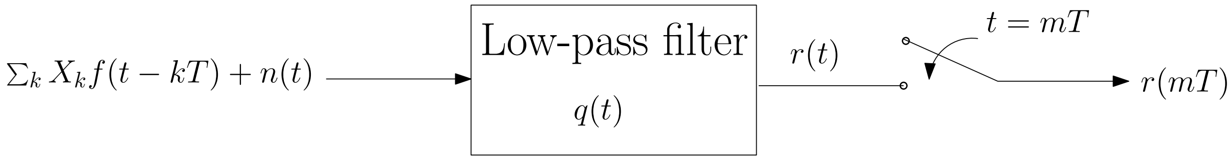
There are two approaches to dealing with ISI:

1. **Channel Equalisation**.
   Equalisation refers to post-processing the demodulator output $\{r(mT)\}_{m \in \mathbb{Z}}$, often using a *digital filter*, to mitigate the effect of ISI and extract the symbols $\{X_m\}$.

2. **Orthogonal Frequency Division Multiplexing (OFDM)**.
   Here, the transmission band is divided into several narrow sub-bands such that over each sub-band the channel response $H(f)$ is roughly constant. Information is then transmitted over the sub-bands using a set of orthogonal sinusoids.

In this handout, we will discuss equalisation, the next one will discuss OFDM.

NOTE: Equalisation is useful to eliminate ISI *regardless* of how it arises. E.g, even if the channel freq. response is flat and Nyquist pulse criterion is satisfied, we will have ISI if our sampling times are not exactly $\{mT\}$.
Equalisation is used to eliminate ISI in such cases.



The output of the demodulator (in the **absence of noise**) is

$$r(mT) = \sum_k X_k\, g((m-k)T), \qquad \text{for } m = 0, 1, \ldots$$
$$= \ldots + X_{m-2}g(2T) + X_{m-1}g(T) + X_m g(0) + X_{m+1}g(-T) + \ldots$$

Defining $r_m = r(mT)$, and $g_m = g(mT)$, we write the above equation as

$$r_m = \sum_\ell g_\ell X_{m-\ell}, \qquad \text{for } m = 0, 1, \ldots. \tag{1}$$

Note that the sequence $\underline{r} = \{r_m\}$ is a convolution of $\underline{X} = \{X_m\}$ and $\underline{g} = \{g_m\}$.

The convolution in Eq. (1) is a sum over all $\ell \in \mathbb{Z}$, but in practice only a finite number of entries in $\underline{g}$ are non-negligible.
(Recall that $g_m = g(mT)$, where $g(t) = p(t) \star h(t) \star q(t)$ is the overall filter.)

For ease of notation, we will assume that the only non-zero entries in $\underline{g}$ are $[g_0, \ldots, g_L] = [g(0), \ldots, g(LT)]$. Hence Eq. (1) becomes

$$r_m = \sum_{\ell=0}^{L} g_\ell X_{m-\ell}, \qquad m = 0, 1, \ldots, \tag{2}$$

Recall from 3F1 that the z-transforms of the sequences $\underline{g}, \underline{X}$, and $\underline{r}$, are:

$$G(z) = \sum_{\ell=0}^{L} g_\ell z^{-\ell}, \quad X(z) = \sum_{\ell=0}^{\infty} X_\ell z^{-\ell}, \quad R(z) = \sum_{\ell=0}^{\infty} r_\ell z^{-\ell}.$$

Then, taking z-transforms on both sides of (2), we obtain

$$R(z) = X(z)G(z). \tag{3}$$

The zero-forcing equaliser

If we pass the demod output $\underline{r}$ through a filter with transfer fn.

$$H_E(z) = \frac{1}{G(z)}, \tag{4}$$

then from (3) and (4), we see that the output of the filter (in the noiseless case) has z-transform

$$R(z)H_E(z) = X(z)G(z) \cdot \frac{1}{G(z)} = X(z),$$

i.e., we recover the sequence of information symbols $\underline{X}$ perfectly!

This equalising filter $H_E(z) = 1/G(z)$ is called the *zero-forcing* filter as it completely eliminates the ISI.

In the presence of noise, the output of the zero-forcing filter is

$$Y(z) = [X(z)G(z) + N(z)]\, H_E(z) = X(z) + N(z)/G(z),$$

or

$$y_m = x_m + \tilde{n}_m, \quad m = 0, 1, \ldots,$$

where the sequence $\{\tilde{n}_m\}$ is the inverse z-transform of $N(z)/G(z)$.

As $G(z) = \sum_{\ell=0}^{L} g_\ell z^{-\ell}$, the transfer function of the zero-forcing filter is

$$H_E(z) = \frac{1}{\sum_{\ell=0}^{L} g_\ell z^{-\ell}} = h_0 + h_1 z^{-1} + h_2 z^{-2} + h_3 z^{-3} + \ldots$$

The key point to note is that the above zero forcing filter is IIR, i.e., impulse response $\underline{h}$ has infinitely many non-zero coefficients.

To see how the the IIR filter can be implemented, observe that the z-transform of the filter output $\underline{y}$ is

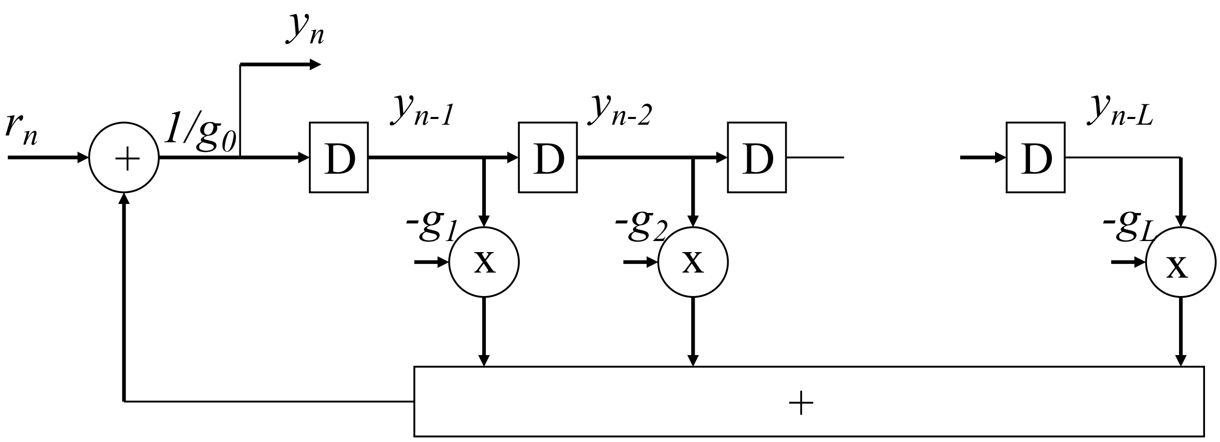
$$Y(z) = H_E(z)R(z) = \frac{R(z)}{G(z)} \quad \Rightarrow \quad R(z) = Y(z)G(z.)$$

Hence

$$\underline{r} = \underline{y} \star \underline{g} \;\Rightarrow r_n = \sum_{\ell=0}^{L} g_\ell y_{n-\ell} \tag{5}$$

We rewrite Eq. (5) as

$$y_n = \frac{1}{g_0}\left[r_n - g_1 y_{n-1} - g_2 y_{n-2} - \ldots - g_L y_{n-L}\right]$$



(D indicates a delay of one sample)

IIR filters are difficult to deal with in practice:

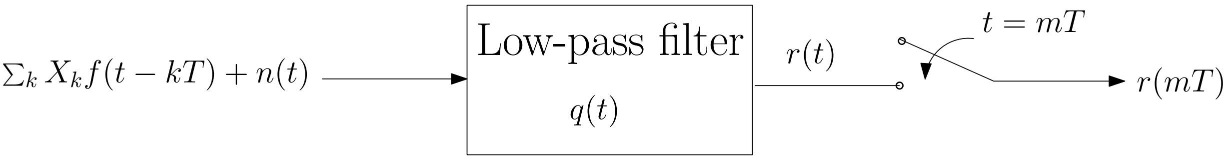
- Stability is not guaranteed
- Their recursive nature also makes them prone to numerical instability

Zero forcing with FIR filters

The easiest approach is to truncate the IIR filter response to a desired number of coefficients. However, this does not always work well.

The better approach is to directly design a zero-forcing FIR filter.

Recall the demodulator output:



$$r_m = \sum_{\ell=0}^{L} g_\ell X_{m-\ell} + n_m, \qquad \text{for } m = 0, 1, \dots. \tag{6}$$

> We want to design an equalising filter with $K+1$ taps, with impulse response $\underline{h} = [h_0, \dots, h_K]$.

If we pass $\underline{r}$ through the filter, the output is $\underline{y} = \underline{h} \star \underline{r}$.

Using (6) (and ignoring the noise variable for the moment) we have for $m = 0, 1, \dots$,

$$\sum_{i=0}^{K} h_i r_{m-i} = \sum_{i=0}^{K} h_i \sum_{\ell=0}^{L} g_\ell X_{m-i-\ell}$$

Using the change of variable $j = i + \ell$, we obtain

$$\sum_{i=0}^{K} h_i r_{m-i} = \sum_{j=0}^{L+K} X_{m-j} \sum_{i=0}^{K} h_i g_{j-i} = \sum_{j=0}^{L+K} X_{m-j} f_j,$$

where $f_j = \sum_{i=0}^{K} h_i g_{j-i}$, for $0 \le j \le L+K$. That is, $\underline{f} = \underline{h} \star \underline{g}$.

> Taking the noise into account, the output of the FIR filter is
>
> $$y_m = X_m f_0 + \sum_{j=1}^{L+K} X_{m-j} f_j + \sum_{i=0}^{K} h_i n_{m-i} \tag{7}$$

For an ideal zero-forcing equaliser, we would like

$$f_0 = 1, \qquad f_j = 0, \text{ for } j = 1, \ldots, L+K,$$

where

$$\begin{aligned} f_0 &= h_0 g_0 \\ f_1 &= h_0 g_1 + g_0 h_1 \\ &\vdots \\ f_K &= h_0 g_K + h_1 g_{K-1} + \ldots + h_K g_0 \\ &\vdots \\ f_{L+K} &= h_K g_L \end{aligned} \tag{8}$$

- We want to simultaneously satisfy these $L+K+1$ equations, but can only specify $K+1$ filter coefficients $h_0, \ldots, h_K$.
- So perfect zero-forcing is not possible with FIR filters.

- The usual approach is to choose $K \geq L$ (number of filter taps at least equal to than number of interfering symbols).
- Then determine $\underline{h} = [h_0, \ldots, h_K]$ by solving

$$f_0 = 1, \quad f_1 = \ldots = f_K = 0.$$

- *Intuition*: In the received symbol

$$r_m = g_0 X_m + g_1 X_{m-1} + \ldots + g_L X_{m-L} + n_m,$$

  typically g_0 has the largest magnitude, g_1 the next largest, and so on. By taking $K \geq L$ and setting $f_1, \ldots, f_K$ to 0, Eq. (8) shows that we cancel out the effect of the strongest interfering coefficients.

Let us illustrate the procedure with an example.

FIR zero-forcing example

Suppose $g_0 = 0.8, g_1 = 0.6, g_2 = 0.3$, and $g_\ell = 0$ for $\ell \notin \{0, 1, 2\}$. That is,

$$r_m = 0.8X_m + 0.6X_{m-1} + 0.3X_{m-2} + n_m$$

Suppose we use a 3-tap filter with impulse response $\underline{h} = [h_0, h_1, h_2]$. To determine h_0, h_1, h_2, use the ZF conditions:

$$\begin{aligned} f_0 &= h_0 g_0 = h_0(0.8) = 1, \\ f_1 &= h_0 g_1 + h_1 g_0 = h_0(0.6) + h_1(0.8) = 0, \\ f_2 &= h_0 g_2 + h_1 g_1 + h_2 g_0 = h_0(0.3) + h_1(0.6) + h_2(0.8) = 0. \end{aligned}$$

Solving these, the filter coefficients are:
$h_0 = 1.25,\ h_1 = -0.9375,\ h_2 = 0.234.$

Using (7), the output of the FIR equaliser is

$$y_m = X_m + f_3 X_{m-3} + f_4 X_{m-4} + \ldots + \sum_{i=0}^{2} h_i n_{m-i},$$

where $f_3 = h_1 g_2 + h_2 g_1 = 0.1409$, and $f_4 = h_2 g_2 = 0.0702$.

Disadvantages of ZF equaliser

The output of the FIR ZF equaliser with $K+1$ taps is

$$\begin{aligned} y_m &= X_m f_0 + \sum_{j=1}^{L+K} X_{m-j} f_j + \sum_{i=0}^{K} h_i n_{m-i} \\ &= X_m + \underbrace{\sum_{j=K+1}^{L+K} X_{m-j} f_j}_{\text{residual interference}} + \underbrace{\sum_{i=0}^{K} h_i n_{m-i}}_{\text{noise}} \end{aligned} \tag{9}$$

Assuming the noise variables $\{n_m\}$ are i.i.d. $\mathcal{N}(0, \sigma^2)$ the output noise variance is

$$\mathbb{E}\Big(\sum_{i=0}^{K} h_i n_{m-i}\Big)^2 = \sigma^2 \Big(\sum_{i=0}^{K} h_i^2\Big).$$

Thus, as the number of filter taps $(K+1)$ increases, we reduce the residual interference, but *increase* the noise variance at the output of the filter. In the previous example, note that $\sum_{i=0}^{2} h_i^2 = 2.496$.

Thus there is a trade-off between minimising residual interference and 'noise enhancement'

MMSE equalisation

The zero forcing approach does not take the noise enhancement effect into account when designing the filter.

The minimum mean squared error (MMSE) equaliser explicitly tries to minimise the expected squared error between X_m and its estimate $\hat{X}_m$.

For a $(K+1)$-tap MMSE equaliser:

- For each m, the estimate $\hat{X}_m$ is generated as:

$$\hat{X}_m = c_0 r_m + c_1 r_{m+1} + \ldots + c_K r_{m+K} \qquad (10)$$

  where $r_m, r_{m+1} \ldots$ are given by (6).
- We want to determine the vector $\underline{c} = [c_0, \ldots, c_K]$ that minimises $\mathbb{E}[(X_m - \hat{X}_m)^2]$.
- Observe from (10) that this equaliser can be implemented as a filter with impulse resp. $[h_0 = c_K, h_1 = c_{K-1}, \ldots, h_K = c_0]$.

Recall that

$$r_m = \sum_{\ell=0}^{L} g_\ell X_{m-\ell} + n_m, \qquad \text{for } m = 0, 1, \ldots.$$

Define:

$$\underline{c} = \begin{bmatrix} c_0 \\ \vdots \\ c_K \end{bmatrix} \quad \text{and } \underline{r} = \begin{bmatrix} r_m \\ \vdots \\ r_{m+K} \end{bmatrix}.$$

We want to choose $\underline{c}$ in order to minimise

$$\mathbb{E}[(X_m - \hat{X}_m)^2] = \mathbb{E}[(X_m - \underline{r}^T \underline{c})^2].$$

The expectation is over the all the constellation symbols that are involved in $\underline{r}$, i.e., $\{X_{m-L}, \ldots, X_{m+K}\}$, and the noise random variables $\{n_m, \ldots, n_{m+K}\}$.

The Orthogonality Principle

For the MMSE estimator, the estimation error $(X_m - \underline{c}^T \underline{r})$ is uncorrelated with the observed data $\underline{r}$.

(We won't prove the orthogonality principle here, but use it to derive the optimal $\underline{c}$)

Deriving the MMSE equaliser

According to the orthogonality principle, the optimal $\underline{c}$ should satisfy

$$\mathbb{E}[r_m(X_m - \underline{r}^T \underline{c})] = 0,$$
$$\mathbb{E}[r_{m+1}(X_m - \underline{r}^T \underline{c})] = 0, \quad \ldots, \quad \mathbb{E}[r_{m+K}(X_m - \underline{r}^T \underline{c})] = 0.$$

This set of equations can be collectively written as

$$\mathbb{E}[\underline{r}(X_m - \underline{r}^T \underline{c})] = \underline{0} \quad \Rightarrow \quad \mathbb{E}[\underline{r}\, X_m] = \mathbb{E}[\underline{r}\,\underline{r}^T]\underline{c}. \qquad (11)$$

Therefore, the MMSE equaliser is: $\underline{c} = \mathbf{R}^{-1}\mathbf{p}$, where

$$\mathbf{R} = \mathbb{E}[\underline{r}\,\underline{r}^T], \quad \mathbf{p} = \mathbb{E}[\underline{r}\, X_m].$$

Note that both the matrix $\mathbf{R}$ and the vector $\mathbf{p}$ are deterministic. Q.9 in Examples paper 2 shows how these can be computed.

Final Remarks

- Both the MMSE and the zero forcing equalisers are *linear* equalisers: the estimate of each symbol is a linear combination of the observed sequence $\{r_m\}$.
- Among linear equalisers, the MMSE equaliser achieves the minimum expected squared error per symbol.
- Note that the same equalisation techniques can be applied to deal with ISI in either baseband channels with PAM or passband channels with QAM. In the latter case, Eq. (6) represents the (complex) demodulator output.
- There are also non-linear equalisation techniques, e.g., "decision feedback equalisers", which we won't discuss here.
- Throughout this handout, we assumed a fixed set of coefficients $\underline{g}$ for the overall filter. In practice, the channel may be time-varying, and the equalisation filter coefficients have to be updated accordingly. This is called "adaptive equalisation" and is widely used in practice, e.g. in telephone networks.