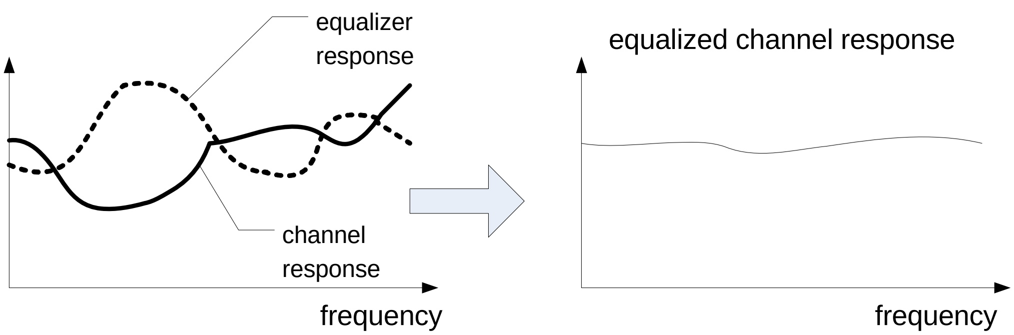
2 Channel Equalization

(Textbook reference: Section 6.6)

Using discrete time representation, the received signal is a filtered and noise-corrupted version of the transmitted sequence:

$$r_k = s_k \otimes c_k + n_k$$

- The multipath channel causes frequency selectivity and ISI
- Equalization can reduce the ISI and noise effects for better demodulation.



Example 1 *In a wireless environment with one direct path and one multipath, the recevied signal is given by*

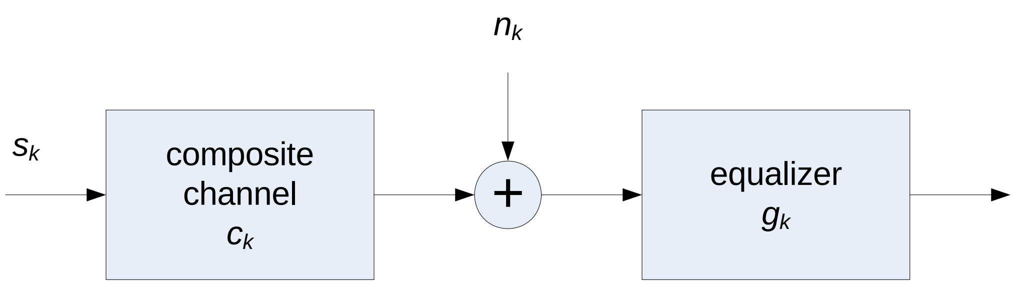
$$r_k = b_1 s_k + b_2 s_{k-l} + n_k$$

The channel response is thus

$$c_k = b_1 \delta(k) + b_2 \delta(k-l)$$

2.1 linear equalization

A linear equalizer is a filter that can *undo* the channel effect



- Ideally, the output of an equalizer is a *delayed* version of the transmitted signal
- A fixed equalizer measures the time-invarient channel and compensates the frequency selectivity during the entire transmission of data
- An adaptive equalizer adjusts its coefficients to track a slowly time-varying channel

Mathematically, the output of an N-tap equalizer $\{g_{0k}, \cdots, g_{Nk}\}$ is given by

$$\widehat{s}_{k-k_o} = \sum_{n=0}^{N} g_{nk} r_{k-n}$$

where r_k is the received signal and k_o here introduces a delay in causal system.

- For fixed equalizers, $g_{nk} = g_n$ (i.e., its coefficients do not change with time k).
- For adaptive equalizers, g_{nk} is updated periodically based on the current channel characteristics.
- The equalizer coefficients can be determined from the channel response, the training sequence, or the data sequence directly

2.2 Zero-forcing

In zero-forcing equalization, the equalizer $\{g_k\}$ attempts to completely inverse the channel by forcing

$$c_k * g_k = \delta(k - k_0)$$

Using matrix representation, this is equivalent to

$$\begin{bmatrix} c_0 & 0 & \cdots & 0 \\ c_1 & c_0 & \ddots & \vdots \\ \vdots & c_1 & \ddots & 0 \\ c_L & \vdots & \ddots & c_0 \\ 0 & c_L & \vdots & c_1 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & c_L \end{bmatrix} \begin{bmatrix} g_0 \\ g_1 \\ \vdots \\ g_N \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix}$$

or

$$\mathbf{Cg} = \mathbf{1}$$

While it is not always possible to find the perfect solution to the above equation (why?), one can approximate it with

$$\mathbf{g} = \mathbf{C}^{\dagger}\mathbf{1}$$

where † denotes the pseudo-inverse (`pinv` in Matlab) of a matrix.

2.3 Adaptive equalization

When the channel is time-varying (LTV), it is necessary to update the equalizer coefficients in order to track the channel changes.

Define the input signal to the equalizer as a vector $\mathbf{r}_k$ where

$$\mathbf{r}_k = [\, r_k \;\; r_{k-1} \;\; r_{k-2} \;\; \ldots \;\; r_{k-N} \,]^T$$

and an equalizer weight vector $\mathbf{g}_k$ where

$$\mathbf{g}_k = [\, g_{0k} \;\; g_{1k} \;\; g_{2k} \;\; \ldots \;\; g_{Nk} \,]^T$$

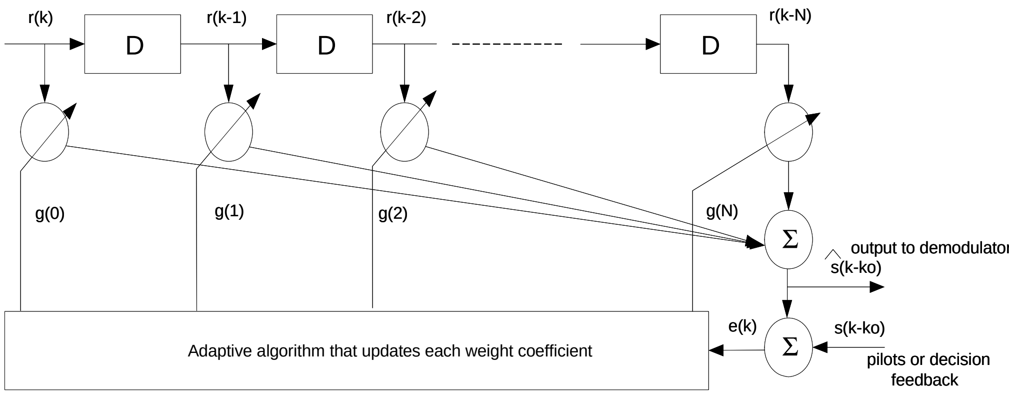
we have

$$\widehat{s}_{k-k_o} = \mathbf{r}_k^T \mathbf{g}_k.$$

The error signal e_k is given by

$$e_k = s_{k-k_o} - \widehat{s}_{k-k_o} = s_{k-k_o} - \mathbf{r}_k^T \mathbf{g}_k$$

An adaptive channel equalizer has the following structure.



In most cases, a *training sequence (or pilot sequence)* is inserted periodically during data transmission. This particular set of symols $\{s_k\}$ are therefore known to the receiver and thus can be used to train the equalizer (i.e., update its cofficients).

The least mean-square (LMS) algorithm carries out the MSE by recursively updating the cofficients using the following rules:

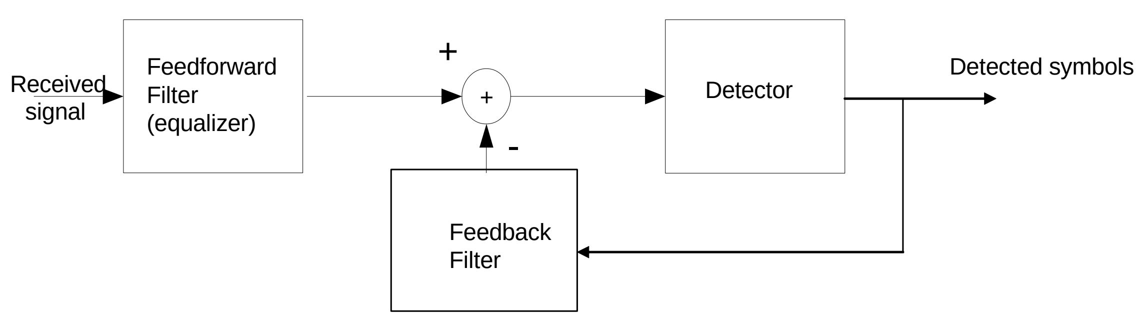
$$\begin{aligned} \widehat{s}_{k-k_o} &= \mathbf{r}_k^T \mathbf{g}_k \\ e_k &= s_{k-k_o} - \widehat{s}_{k-k_o} \\ \mathbf{g}_{k+1} &= \mathbf{g}_k + \alpha e_k^* \mathbf{r}_k \end{aligned}$$

- The step parameter, α, controls the adaptation rate and must be chosen carefully to guarantee convergence.

- The equalizer is converged if the error e_k becomes steady.

2.4 Decision-feedback equalization

- Since the training sequence carries no data information, it incurs an overhead to wireless communications, especially in mobile applications where the equalizer coefficients need to be updated frequently.
- Notice that s_{k-k_o} can be either the pilot symbol or the feedback from a data symbol decision. A decision-feedback equalizer (DFE) is a nonlinear equalizer that employs previous decisions as training sequences.
- The detected symbols (or the output of the feedback filter) is subtracted from the output of the equalizer for adaptation
- While better in performance in general, a major drawback of DFE is its potential catastrophic behavior due to error propagation
- In practical, DFE is often combined with training sequence-based equalization for robustness



2.5 Software Design and Implementation Flow