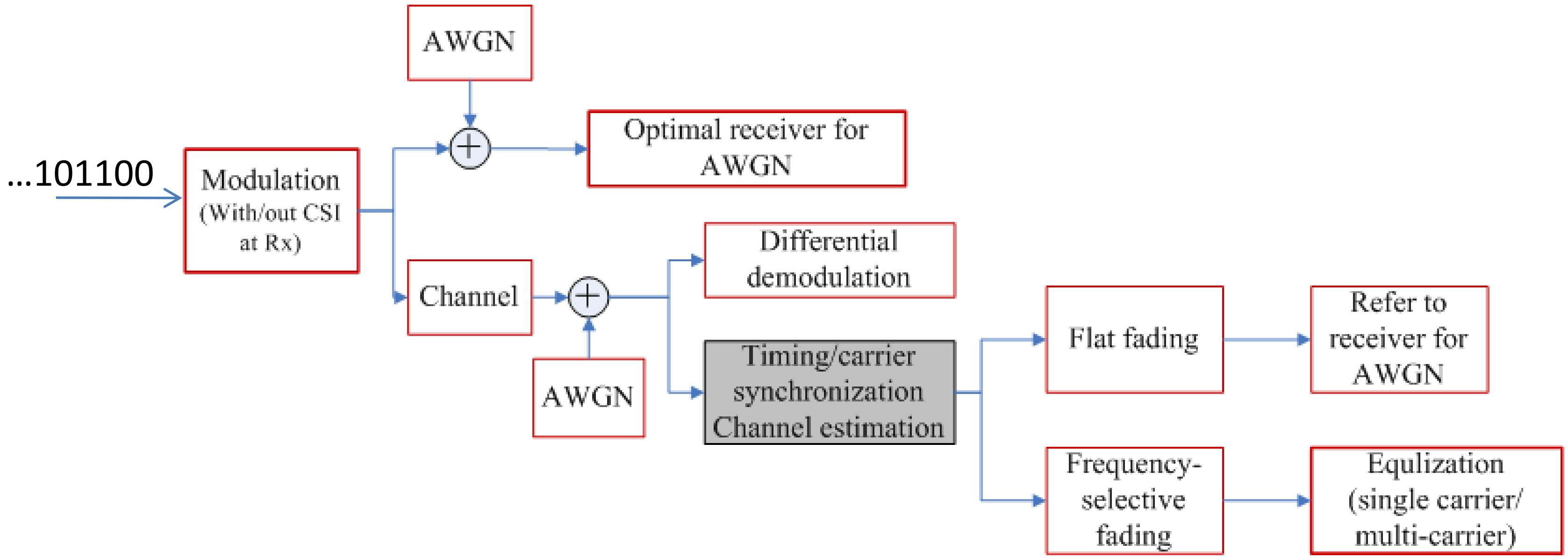
Principle of Digital Communication

---Lecture 4

Part II Channel equalization

4.0 About channel



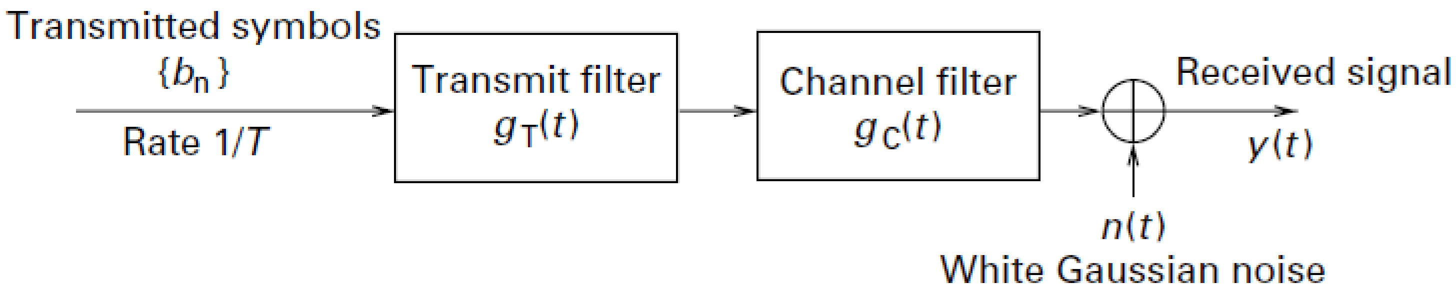
4.1 Introduction

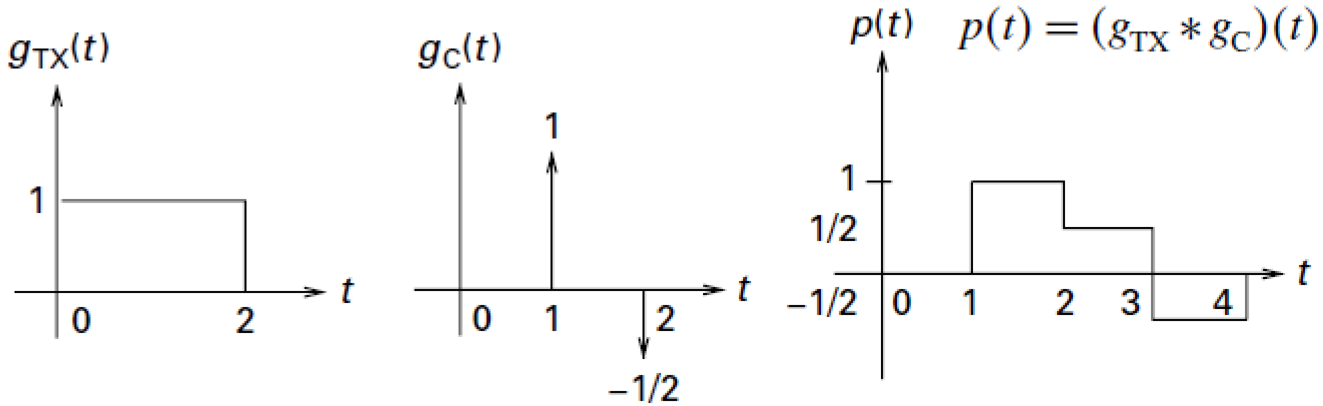
In this chapter, we develop *channel equalization* techniques for handling the intersymbol interference (ISI) incurred by a linearly modulated signal that goes through a dispersive channel.
The techniques developed in this chapter apply to single-carrier systems.

An alternative technique for handling dispersive channels is the use of multicarrier modulation, or orthogonal frequency division multiplexing (OFDM). Roughly speaking, OFDM, or multicarrier modulation, transforms a system with memory into a memoryless system in the frequency domain, by decomposing the channel into parallel narrowband subchannels, each of which sees a scalar channel gain.

4.2 Dispersive channel and Eye diagrams

Consider the complex baseband model for linear modulation over a dispersive channel, as depicted in the following figure.

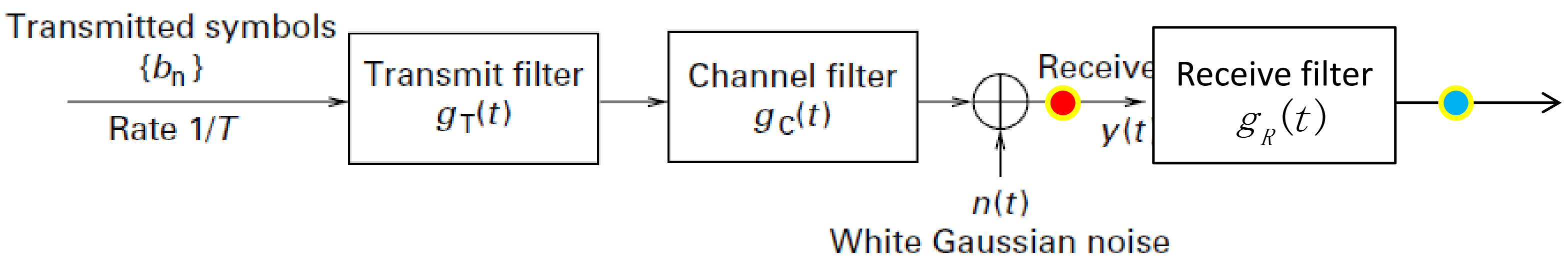




Eye diagrams

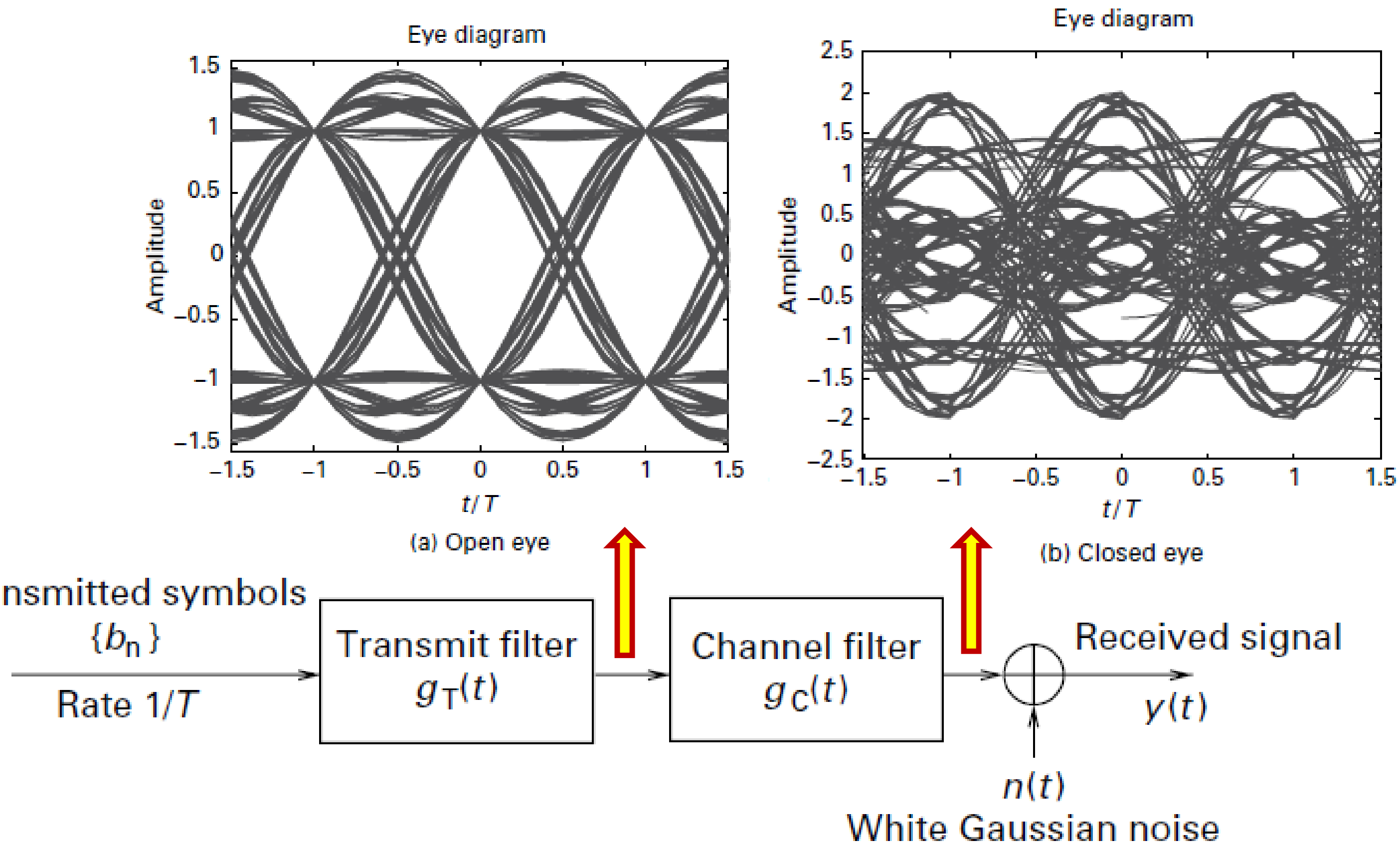
Consider the noiseless signal $r(t) = \sum_n b[n]x(t - nT)$ where $\{b[n]\}$ is the transmitted symbol sequence. The waveform x(t) is the effective symbol waveform.

For an eye diagram at the input to the receive filter, it is the cascade of the transmit and channel filters;
For an eye diagram at the output of the receive filter, it is the cascade of the transmit, channel, and receive filters.



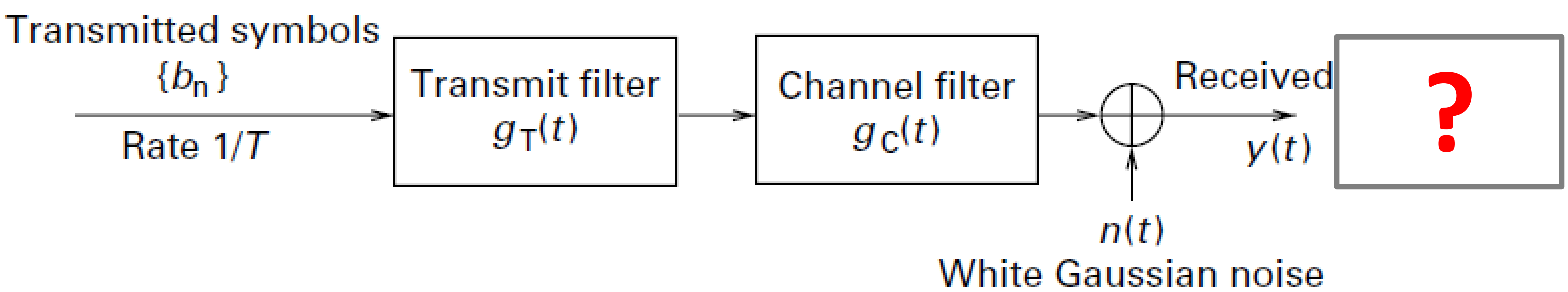
The effect of ISI seen by different symbols is different, depending on how the contributions due to neighboring symbols add up. The eye diagram puts the ISI patterns seen by different symbols into one plot.

Let's see the eye diagram at the input to the receive filter for BPSK using a raised cosine pulse with 50% excess bandwidth as an example. The interval is 3T.



4.3 The optimal receive filter

Consider the complex baseband model for linear modulation over a dispersive channel, as depicted in the following figure.



The signal sent over the channel is given by

$$u(t) = \sum_{n=-\infty}^{\infty} b[n] g_{TX}(t - nT),$$

where $g_{TX}(t)$ is the impulse response of the transmit filter, and b[n] is the symbol sequence, transmitted at rate 1/T. The channel is modeled as a filter with impulse response $g_C(t)$, followed by AWGN. Thus, the received signal is given by

$$y(t) = \sum_{n=-\infty}^{\infty} b[n] p(t - nT) + n(t), \quad p(t) = (g_{TX} * g_C)(t)$$

The task of the channel equalizer is to extract the transmitted sequence $\mathbf{b} = \{b[n]\}$ from the received signal $y(t)$.

Theorem 4.1 (Optimality of the matched filter)
The optimal receive filter is matched to the equivalent pulse p(t), and is specified in the time and frequency domains as follows:

$$\begin{aligned} g_{\mathrm{R,opt}}(t) &= \underline{p_{\mathrm{MF}}(t) = p^*(-t)}, \\ G_{\mathrm{R,opt}}(f) &= P_{\mathrm{MF}}(f) = P^*(f). \end{aligned} \tag{4.1}$$

In terms of a decision on **b**, there is no loss of relevant information to sample the matched filter output at symbol rate

$$z[n] = (y * p_{\mathrm{MF}})(nT) = \int y(t)\underline{p_{\mathrm{MF}}(nT-t)}\,\mathrm{d}t = \int y(t)\underline{p^*(t-nT)}\,\mathrm{d}t. \tag{4.2}$$

Proof of Theorem 4.1 Deciding on the sequence $\mathbf{b}$ is equivalent to testing between all possible hypothesized sequences $\mathbf{b}$, with the hypothesis $H_{\mathbf{b}}$ corresponding to sequence $\mathbf{b}$ given by

$$H_{\mathbf{b}} : y(t) = s_{\mathbf{b}}(t) + n(t),$$

where

$$s_{\mathbf{b}}(t) = \sum_n b[n]p(t - nT)$$

is the noiseless received signal corresponding to transmitted sequence $\mathbf{b}$. We know from Theorem 3.4.3 that the ML rule is given by

$$\delta_{\text{ML}}(y) = \arg\max_{\mathbf{b}} \ \text{Re}(\langle y, s_{\mathbf{b}}\rangle) - \frac{||s_{\mathbf{b}}||^2}{2}.$$

The MPE rule is similar, except for an additive correction term accounting for the priors. In both cases, the decision rule depends on the received signal only through the term $\langle y, s_{\mathbf{b}}\rangle$. The optimal front end, therefore, should capture enough information to be able to compute this inner product for all possible sequences $\mathbf{b}$.

Let us now consider the structure of this inner product in more detail.

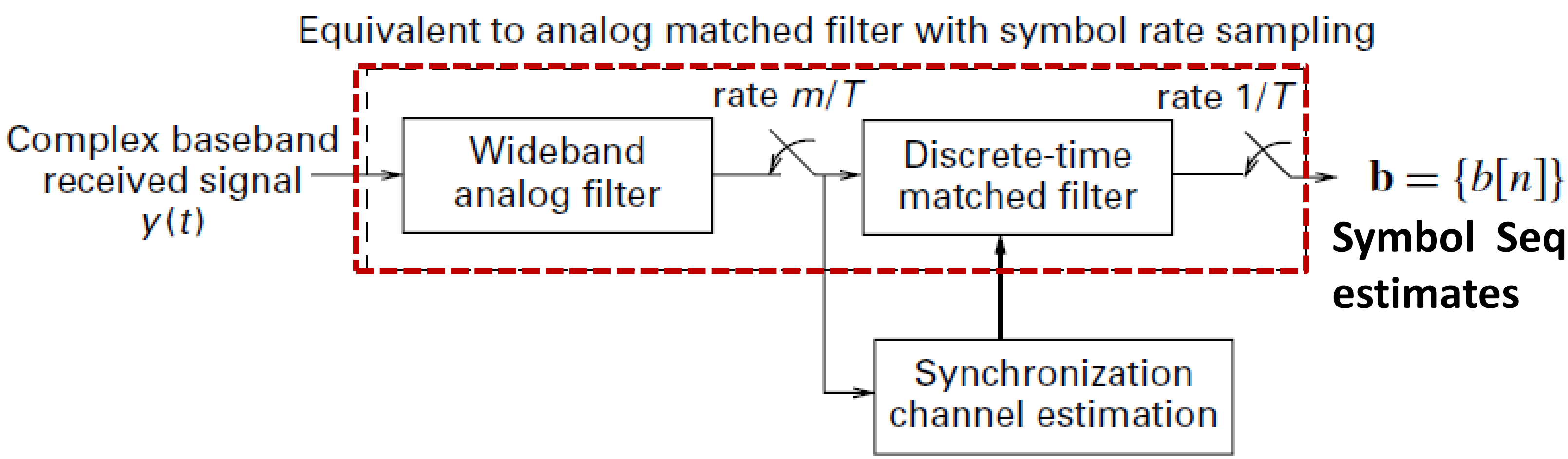
$$\langle y, s_{\mathrm{b}}\rangle = \langle y, \sum_n b[n]p(t-nT)\rangle = \sum_n b^*[n] \underline{\int y(t)p^*(t-nT)\,\mathrm{d}t} = \sum_n b^*[n]\underline{z[n]},$$

where $\{z[n]\}$ are as in (4.2). Generation of $\{z[n]\}$ by sampling the outputs of the matched filte (4.1) at the symbol rate follows immediately from the definition of the matched filter. □

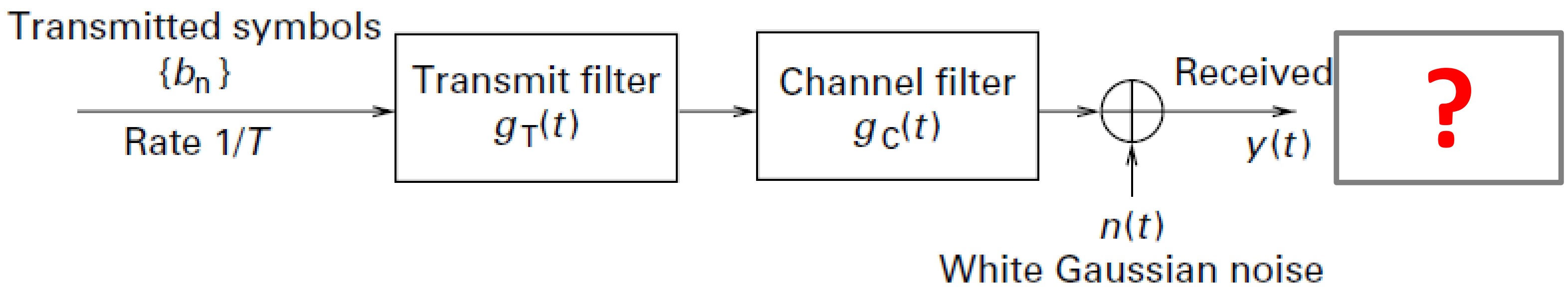
$$z[n] = (y * p_{\mathrm{MF}})(nT) = \int y(t)\underline{p_{\mathrm{MF}}(nT-t)}\,\mathrm{d}t = \int y(t)\underline{p^*(t-nT)}\,\mathrm{d}t. \quad (4.2)$$

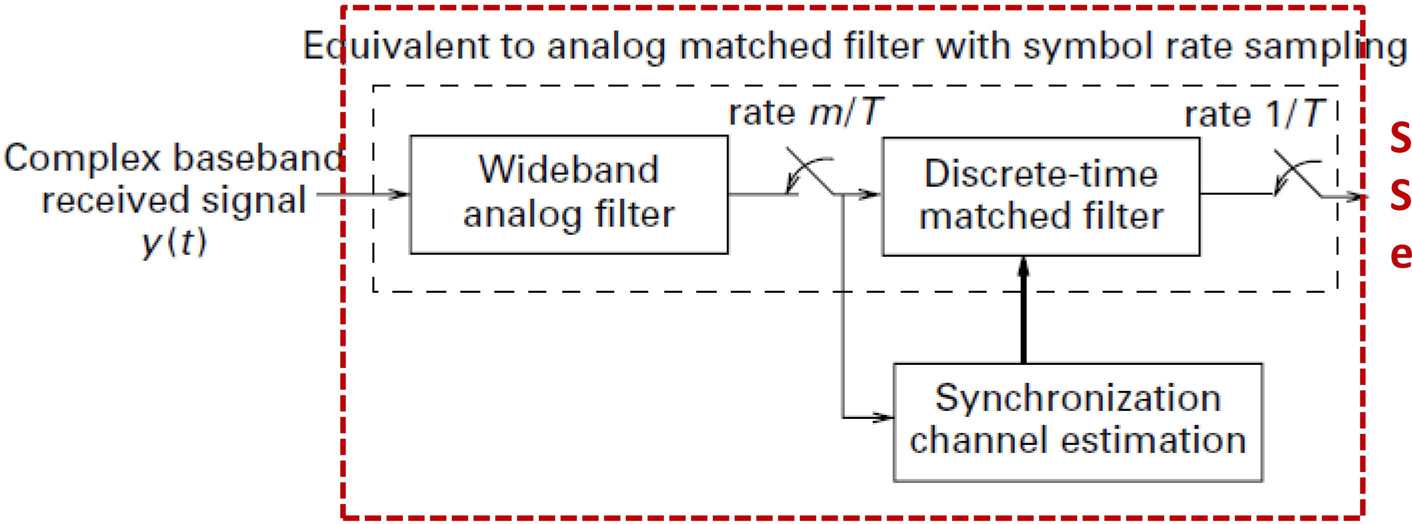
$$\boxed{\begin{aligned} g_{\mathrm{R,opt}}(t) &= p_{\mathrm{MF}}(t) = p^*(-t), \\ G_{\mathrm{R,opt}}(f) &= P_{\mathrm{MF}}(f) = P^*(f). \end{aligned}} \quad (4.1)$$

A typical implementation of the matched filter is in discrete time domain. The matched filter is used after estimating the effective discrete time channel (typically using a sequence of known training symbols) .



Symbol Sequence estimates





Symbol Sequence estimates

4.4 Maximum likelihood sequence estimation

We develop a method for ML estimation of the entire sequence $\mathbf{b} = \{b[n]\}$ based on the received signal model

$$y(t) = \sum_{n=-\infty}^{\infty} b[n]p(t-nT) + n(t), \ \ p(t) = (g_{\mathrm{TX}} * g_{\mathrm{C}})(t)$$

Theorem 4.1 tells us that the optimal front end is the filter matched to the cascade of the transmit and channel filters.

Deciding on the sequence **b** is equivalent to testing between all possible hypothesized sequences, the hypothesis $H_{\mathbf{b}}$ is corresponding to sequence **b** :

$$H_{\mathbf{b}} : y(t) = s_{\mathbf{b}}(t) + n(t), \ \ s_{\mathbf{b}}(t) = \sum_{n} b[n]p(t-nT)$$

the ML rule is given by

$$\delta_{\mathrm{ML}}(y) = \arg \max_{\mathbf{b}} \ \mathrm{Re}(\langle y, s_{\mathbf{b}} \rangle) - \frac{||s_{\mathbf{b}}||^2}{2}.$$

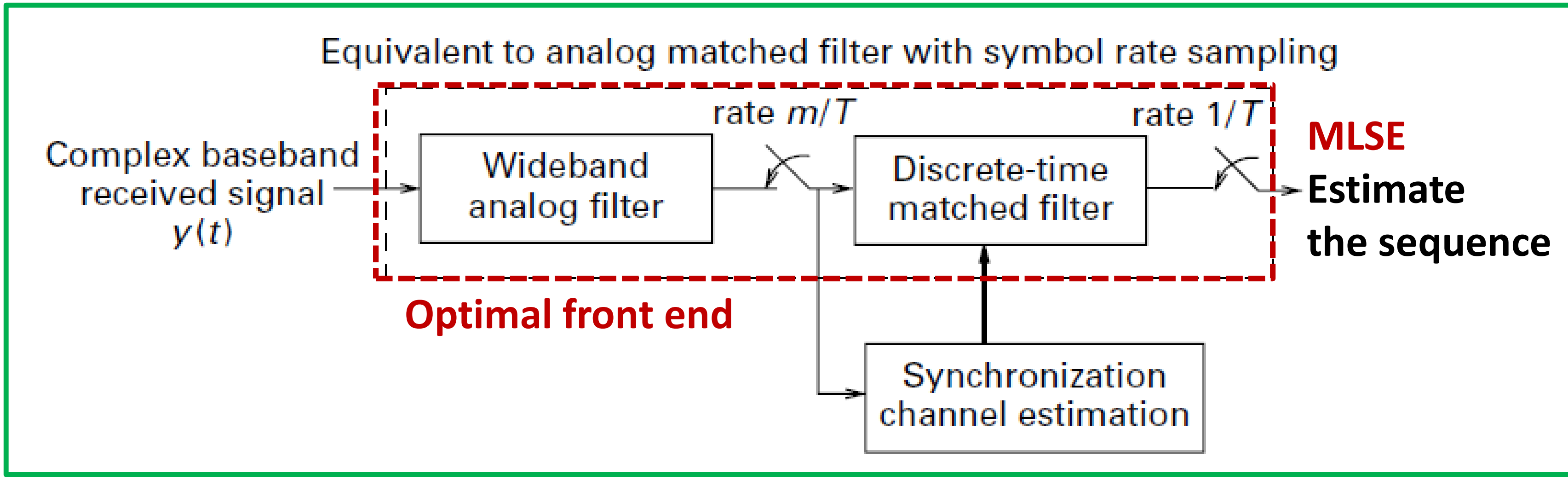
$$\delta_{\mathrm{ML}}(y) = \arg\max_{\mathbf{b}} \operatorname{Re}(\langle y, s_{\mathbf{b}} \rangle) - \frac{||s_{\mathbf{b}}||^2}{2}.$$

Equivalently, over all possible sequences b we wish to maximize

$$\Lambda(\mathbf{b}) = \operatorname{Re}\langle y, s_{\mathbf{b}} \rangle - \frac{||s_{\mathbf{b}}||^2}{2} \qquad s_{\mathbf{b}}(t) = \sum_n b[n]p(t - nT)$$

Suppose that N symbols are sent, each drawn from an M-ary alphabet. Then there are M^N possible sequences **b** that must be considered in the maximization, the complexity grows quickly for any reasonable sequence length (e.g., direct ML estimation for 1000 QPSK symbols incurs a complexity of 4^{1000}).

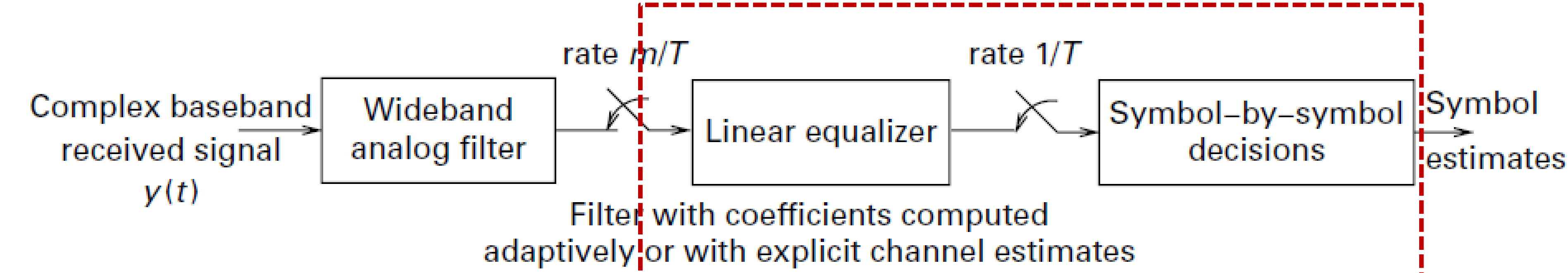
We would like to develop a form for the cost function that we can compute simply by adding terms as we increase the symbol time index n.
--to reduce the complexity by using Viterbi algorithm (has complexity $O(M^L)$ per demodulated symbol, L is the number of multi-paths)



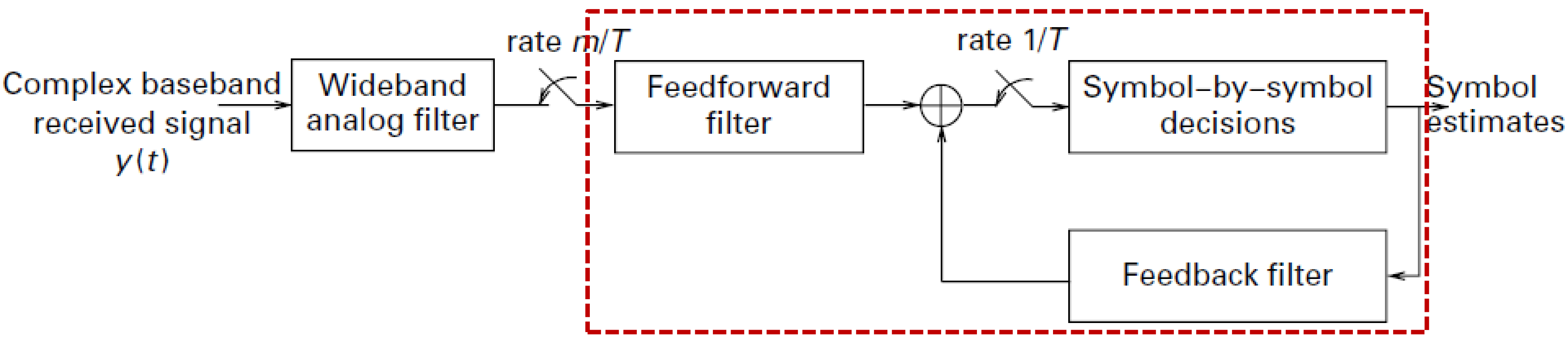
The complexity of the optimum MLSE receiver may be too high for large constellations or large channel memory.

We will consider suboptimal equalization strategies whose complexity scales linearly with the channel memory. The schemes we describe can be used to adaptive implementation, and do not require an optimal front end.

4.5 Suboptimal equalizer design



Linear equalization



Decision feedback equalization

Assume that the received signal is passed through an arbitrary receive filter $g_{\mathrm{RX}}(t)$, and sampled at a rate $1/T_s = m/T$, where m is a positive integer: m = 1 corresponds to *symbol spaced sampling*, while m>1 corresponds to *fractionally spaced sampling.* The received signal is given by

$$y(t) = \sum_n b[n]p(t - nT) + n(t)$$

The output of the sampler is a discrete-time sequence r[k], where

$$r[k] = (y * g_{\mathrm{RX}})(kT_s + \delta)$$

sampling offset

Let's consider the signal and noise components separately.

First we consider the response at the output of the sampler, to a single symbol, say b[0]. This is given by the discrete-time impulse response

$$f[k] = (p * g_{\mathrm{RX}})(kT_s + \delta), \qquad k = \ldots, -1, 0, 1, 2, \ldots$$

The next symbol sees the same response, shifted by the symbol interval T, which corresponds to m samples, and so on.

The noise sequence at the output of the sampler is given by

$$w[k] = (n * g_{\mathrm{RX}})(kT_s + \delta).$$

If n is complex WGN, the noise at the receive filter output, $w(t) = (n * g_{\mathrm{RX}})(t)$ is zero mean, proper[1] complex, Gaussian random. The sampled noise sequence {w[k]} is zero mean, proper complex Gaussian, with autocovariance function

$$C_{\mathrm{w}}(k) = \mathrm{cov}(w_{n+k}, w_n) = 2\sigma^2 \int g_{\mathrm{RX}}(t) g^*_{\mathrm{RX}}(t - kT_s)\mathrm{d}t.$$

$$\begin{aligned} \mathbf{C}_X(i,j) = \mathrm{cov}(X_i, X_j) &= \mathbb{E}[(X_i - \mathbb{E}[X_i])(X_j - \mathbb{E}[X_j])] \\ &= \mathbb{E}[X_i X_j] - \mathbb{E}[X_i]\mathbb{E}[X_j]. \end{aligned}$$

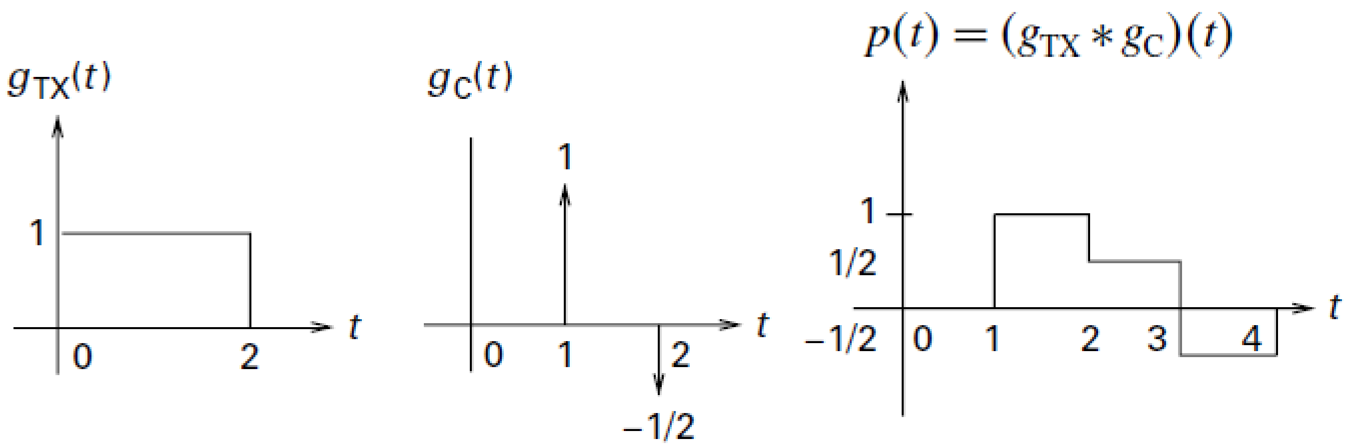
[1]: **Proper:** A complex RV $\mathbf{z}$ is called proper if $\widetilde{\mathbf{C}}_{\mathbf{z}} = \mathbf{0}$; otherwise, it is called improper.

Zero-mean complex-valued random vector (RV): $\mathbf{z} = \mathbf{u} + j\mathbf{v}$

Covariance matrix: $\mathbf{C}_{\mathbf{z}} \triangleq \mathbb{E}(\mathbf{z}\mathbf{z}^H)$

Pseudo-covariance matrix: $\widetilde{\mathbf{C}}_{\mathbf{z}} \triangleq \mathbb{E}(\mathbf{z}\mathbf{z}^T)$

Running example As a running example through this chapter, we consider the setting shown as follows. . The symbol rate is 1/2 (i.e., one symbol every two time units). The transmit pulse $g_{TX}(t) = I_{[0,2]}(t)$ is an ideal rectangular pulse in the time domain, while the channel response $g_C(t) = \delta(t-1) - 1/2\delta(t-2)$ corresponds to two discrete paths.



consider a receive filter $g_{RX}(t) = I_{[0,1]}$. Note that this receive filter in this example is not matched to either the transmit filter or to the cascade of the transmit filter and the channel. The symbol interval $T = 2$, and we choose a sampling interval $T_s = 1$; that is, we sample twice as fast as the symbol rate. Note that the impulse response of the receive filter is

To define the noise contribution, note that the autocovariance function of the complex Gaussian noise samples is given by

$$C_{\mathrm{w}}[k] = 2\sigma^2 \delta_{k0}.$$

That is, the noise samples are complex WGN. Suppose, now, that we wish to make a decision on the symbol $b[n]$ based on a block of five samples $\mathbf{r}[n]$, chosen such that $b[n]$ makes a strong contribution to the block. The model for such a block can be written as

$$\mathbf{r}[n] = b[n-1]\begin{pmatrix} \frac{1}{2} \\ -\frac{1}{2} \\ 0 \\ 0 \\ 0 \end{pmatrix} + b[n]\begin{pmatrix} 0 \\ 1 \\ \frac{1}{2} \\ -\frac{1}{2} \\ 0 \end{pmatrix} + b[n+1]\begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ \frac{1}{2} \end{pmatrix} + \mathbf{w}_n = \mathbf{U}\mathbf{b}[n] + \mathbf{w}[n],$$

where $\mathbf{w}[n]$ is discrete-time WGN,

$$\mathbf{b}[n] = \begin{pmatrix} b[n-1] \\ b[n] \\ b[n+1] \end{pmatrix}$$

is the block of symbols making a nonzero contribution to the block of samples, and

$$\begin{array}{cc} & \begin{array}{ccc} \phantom{-\frac{1}{2}} & \mathbf{u}_0 & \mathbf{u}_i \end{array} \quad \textbf{i= -1,1} \\ \mathbf{U} = & \begin{pmatrix} \frac{1}{2} & 0 & 0 \\ -\frac{1}{2} & 1 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & -\frac{1}{2} & 1 \\ 0 & 0 & \frac{1}{2} \end{pmatrix} \end{array}$$

is a matrix whose columns equal the responses corresponding to the symbols contributing to $\mathbf{r}[n]$. The middle column corresponds to the *desired* symbol $b[n]$, while the other columns correspond to the interfering symbols $b[n-1]$ and $b[n+1]$. The columns are acyclic shifts of the basic discrete impulse response to a single symbol, with the entries shifting down by one symbol interval (two samples in this case) as the symbol index is incremented. We use $\mathbf{r}[n]$ to decide on $b[n]$

Previous symbol Current symbol Future symbol

$$\mathbf{r}[n] = b[n-1]\begin{pmatrix} \frac{1}{2} \\ -\frac{1}{2} \\ 0 \\ 0 \\ 0 \end{pmatrix} + b[n]\begin{pmatrix} 0 \\ 1 \\ \frac{1}{2} \\ -\frac{1}{2} \\ 0 \end{pmatrix} + b[n+1]\begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ \frac{1}{2} \end{pmatrix} + \mathbf{w}_n = \mathbf{U}\mathbf{b}[n] + \mathbf{w}[n],$$

n=n+1

$$\mathbf{r}[n+1] = b[n]\begin{pmatrix} \frac{1}{2} \\ -\frac{1}{2} \\ 0 \\ 0 \\ 0 \end{pmatrix} + \underset{b[n+1]}{b[n]}\begin{pmatrix} 0 \\ 1 \\ \frac{1}{2} \\ -\frac{1}{2} \\ 0 \end{pmatrix} + b[n+2]\begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ \frac{1}{2} \end{pmatrix} + \mathbf{w}_n = \mathbf{U}\mathbf{b}[n+1] + \mathbf{w}[n+1]$$

For a decision on the next symbol, $b[n+1]$, we simply shift the window of samples to the right by a symbol interval (i.e., by two samples), to obtain a vector $\mathbf{r}[n+1]$. Now $b[n+1]$ becomes the desired symbol, and $b[n]$ and b_{n+2} the interfering symbols, but the basic model remains the same.

End of the example

The model for the received vector is

$$\mathbf{r}[n] = \underset{L \times K}{\mathbf{U}} \; \underset{K \times 1}{\mathbf{b}[n]} + \mathbf{w}[n], \qquad K = k_1 + k_2 + 1.$$

$$\mathbf{b}[n] = (\underline{b[n-k_1], \ldots, b[n-1]}, b[n], \underline{b[n+1], \ldots, b[n+k_2]})^T$$

The columns of **U** is the responses corresponding to the individual symbols. All of these column vectors are acyclic shifts of the basic discrete-time impulse response to a single symbol, which is given by the samples of

$$g_{\mathrm{TX}} * g_{\mathrm{C}} * g_{\mathrm{RX}}$$

The noise vector **w**[n] is zero mean, proper complex Gaussian with covariance matrix $\mathbf{C}_{\mathrm{w}}$

[1]: **Proper:** A complex RV $\mathbf{z}$ is called proper if $\widetilde{\mathbf{C}}_{\mathbf{z}} = \mathbf{0}$; otherwise, it is called improper.

Zero-mean complex-valued random vector (RV): $\mathbf{z} = \mathbf{u} + j\mathbf{v}$

Covariance matrix: $\mathbf{C}_{\mathbf{z}} \triangleq \mathbb{E}(\mathbf{z}\mathbf{z}^H)$

Pseudo-covariance matrix: $\widetilde{\mathbf{C}}_{\mathbf{z}} \triangleq \mathbb{E}(\mathbf{z}\mathbf{z}^T)$

4.6 Linear equalization

Linear equalization corresponds to correlating **r**[n] with a vector **c** to produce a decision statistic $Z[n] = \langle \mathbf{r}[n], \mathbf{c} \rangle = \mathbf{c}^H \mathbf{r}[n]$
This decision statistic is then employed to generate either hard or soft decisions for b[n]. Rewriting **r**[n] as

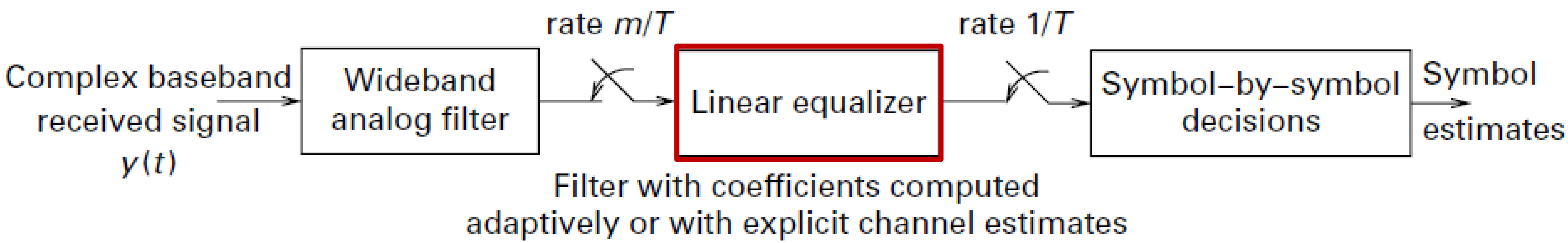
$$\mathbf{r}[n] = b[n]\mathbf{u}_0 + \sum_{i \neq 0} b[n+i]\mathbf{u}_i + \mathbf{w}[n],$$

$$\Downarrow \quad Z[n] = \langle \mathbf{r}[n], \mathbf{c} \rangle = \mathbf{c}^H \mathbf{r}[n]$$

$$Z[n] = \mathbf{c}^H \mathbf{r}[n] = b[n](\mathbf{c}^H \mathbf{u}_0) + \sum_{i \neq 0} b[n+i](\mathbf{c}^H \mathbf{u}_i) + \mathbf{c}^H \mathbf{w}[n].$$

To make a reliable decision on b[n] based on Z[n], we must choose **c** such that the term $\mathbf{c}^H \mathbf{u}_0$ is significantly larger than the “residual ISI” terms $\mathbf{c}^H \mathbf{u}_i$, $i \neq 0$. We must also keep in mind the effects of the noise term $\mathbf{c}^H \mathbf{w}[n]$, which is zero mean Gaussian with covariance $\mathbf{c}^H \mathbf{C}_\mathrm{w} \mathbf{c}$.

The correlator **c** can also be implemented as a discrete-time filter, whose outputs are sampled at the symbol rate to obtain the desired decision statistics Z[n]. Such an architecture is depicted in the following figure.



A typical architecture for implementing a linear equalizer

Zero-forcing (ZF) equalizer

Linear MMSE equalizer

Zero-forcing (ZF) equalizer

The ZF equalizer addresses the preceding considerations by insisting that the ISI at the correlator output be set to zero. The ZF solution, if it exists, satisfies

$$Z[n] = \mathbf{c}^H \mathbf{r}[n] = b[n](\mathbf{c}^H \mathbf{u}_0) + \sum_{i \neq 0} b[n+i](\mathbf{c}^H \mathbf{u}_i) + \mathbf{c}^H \mathbf{w}[n].$$

$$\mathbf{c}^H \mathbf{u}_0 = 1, \quad \mathbf{c}^H \mathbf{u}_i = 0, \quad \text{for all } i \neq 0.$$

$$\mathbf{c}^H \mathbf{U} = (0, \ldots, 0, 1, 0, \ldots, 0) = \mathbf{e}^T, \text{ or } \mathbf{U}^H \mathbf{c} = \mathbf{e}.$$

where the nonzero entry on the right-hand side corresponds to the column with the desired signal vector.

The problem can be formulated as

minimize $||\mathbf{c}||^2$

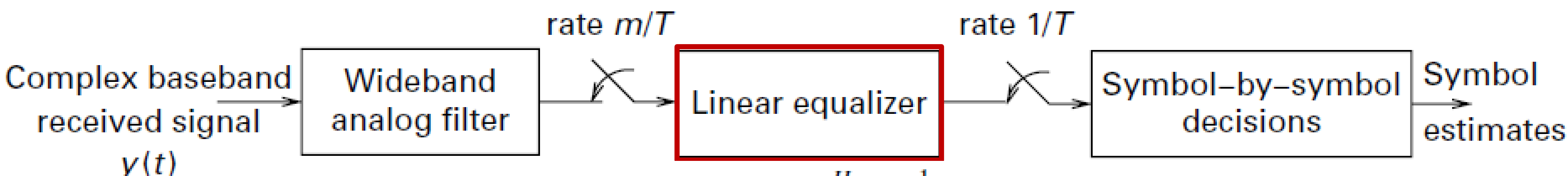
subject to $\mathbf{U}^H\mathbf{c} = \mathbf{e}$

To minimize $||\mathbf{c}||^2$ subject to $\mathbf{U}^H\mathbf{c} = \mathbf{e}$, any component orthogonal to the subspace spanned by the signal vectors $\{\mathbf{u}_i\}$ should be set to zero, so that we may insist that **c** is a linear combination of the **u**i, given by $\mathbf{c} = \mathbf{U}\mathbf{a}$, where the K×1 vector **a** contains the coefficients of the linear combination. Substituting in $\mathbf{U}^H\mathbf{c} = \mathbf{e}$ we obtain

$$\mathbf{U}^H\mathbf{U}\mathbf{a} = \mathbf{e}. \Longrightarrow \left.\begin{array}{l} \mathbf{a} = (\mathbf{U}^H\mathbf{U})^{-1}\mathbf{e}, \\ \mathbf{c} = \mathbf{U}\mathbf{a}, \end{array}\right\} \mathbf{c}_{\mathrm{ZF}} = \mathbf{U}(\mathbf{U}^H\mathbf{U})^{-1}\mathbf{e}.$$

$$\mathbf{r}[n]$$

$$Z[n] = \langle \mathbf{r}[n], \mathbf{c} \rangle = \mathbf{c}^H \mathbf{r}[n].$$



$$\mathbf{c}_{\mathrm{ZF}} = \mathbf{U}(\mathbf{U}^H\mathbf{U})^{-1}\mathbf{e}.$$

$$b[n]$$

$$\mathbf{r}[n] = b[n]\mathbf{u}_0 + \sum_{i \neq 0} b[n+i]\mathbf{u}_i + \mathbf{w}[n]$$

$$Z[n] = \mathbf{c}^H\mathbf{r}[n] = b[n](\mathbf{c}^H\mathbf{u}_0) + \cancel{\sum_{i \neq 0} b[n+i](\mathbf{c}^H\mathbf{u}_i)} + \mathbf{c}^H\mathbf{w}[n].$$

$$\langle \mathbf{c}, \mathbf{u}_0 \rangle = 1$$

The noise enhancement factor is the price we pay for knocking out the ISI.

Linear MMSE equalizer

The design of the ZF equalizer ignores the effect of noise at the equalizer output.

An alternative to this is the linear minimum mean squared error (MMSE) criterion, which trades off the effect of noise and ISI at the equalizer output. The mean squared error (MSE) at the output of a linear equalizer **c** is defined as

$$\mathrm{MSE} = J(\mathbf{c}) = \mathbb{E}[|\mathbf{c}^H \mathbf{r}[n] - b[n]|^2],$$

where the expectation is taken over the symbol stream $\{b[n]\}$. The MMSE equalizer is given by

$$\mathbf{c}_{\mathrm{MMSE}} = \mathbf{R}^{-1}\mathbf{p},$$

$$\mathbf{R} = \mathbb{E}[\mathbf{r}[n](\mathbf{r}[n])^H], \quad \mathbf{p} = \mathbb{E}[b^*[n]\mathbf{r}[n]].$$

The MMSE criterion is useful in many settings, not just equalization.

The MMSE equalizer can be proved using the orthogonality principle.

Proof using the orthogonality principle

Suppose that we wish to find the best linear approximation for a complex random variable Y in terms of a sequence of complex random variables $\{X_i\}$. Thus, the approximation must be of the form $\sum_i a_i X_i$, where $\{a_i\}$ are complex scalars to be chosen to minimize the MSE:

$$\mathbb{E}\left[|Y - \sum_i a_i X_i|^2 \right].$$

The preceding can be viewed as minimizing a distance in a space of random variables, in which the inner product is defined as

$$\langle U, V \rangle = \mathbb{E}[UV^*].$$

This satisfies all the usual properties of an inner product: $\langle aU, bV \rangle = ab^* \langle U, V \rangle$, and $\langle U, U \rangle = 0$ if and only if $U = 0$ (where equalities are to be interpreted as holding with probability one). The orthogonality principle holds for very general inner product spaces, and states that, for the optimal approximation, the approximation error is orthogonal to every element of the approximating space. Specifically, defining the error as

$$e = Y - \sum_i a_i X_i,$$

$$e = Y - \sum_i a_i X_i,$$

we must have

$$\langle X_i, e \rangle = 0 \quad \text{for all } i.$$

Applying it to our setting, we have $Y = b[n]$, X_i are the components of $\mathbf{r}[n]$, and

$$e = \mathbf{c}^H \mathbf{r}[n] - b[n].$$

In this setting, the orthogonality principle can be compactly stated as

$$0 = \mathbb{E}[\mathbf{r}[n]e^*] = \mathbb{E}[\mathbf{r}[n]((\mathbf{r}[n])^H \mathbf{c} - b^*[n])] = \mathbf{R}\mathbf{c} - \mathbf{p}.$$

This completes the proof. □

$$\mathbf{c}_{\text{MMSE}} = \mathbf{R}^{-1}\mathbf{p},$$

$$\mathbf{R} = \mathbb{E}[\mathbf{r}[n](\mathbf{r}[n])^H], \qquad \mathbf{p} = \mathbb{E}[b^*[n]\mathbf{r}[n]].$$

$$\mathbf{r}[n] = \mathbf{U}\ \mathbf{b}[n] + \mathbf{w}[n] = b[n]\mathbf{u}_0 + \sum_{i \neq 0} b[n+i]\mathbf{u}_i + \mathbf{w}[n],$$

Assuming that the **symbols** $\{b[n]\}$ are uncorrelated, with $\mathbb{E}[b[n]b^*[m]] = \sigma_b^2 \delta_{nm}$
Then

$$\boxed{\begin{aligned} \mathbf{c}_{\mathrm{MMSE}} &= \mathbf{R}^{-1}\mathbf{p}, \\ \mathbf{R} = \mathbb{E}[\mathbf{r}[n](\mathbf{r}[n])^H], &\quad \mathbf{p} = \mathbb{E}[b^*[n]\mathbf{r}[n]]. \end{aligned}}$$

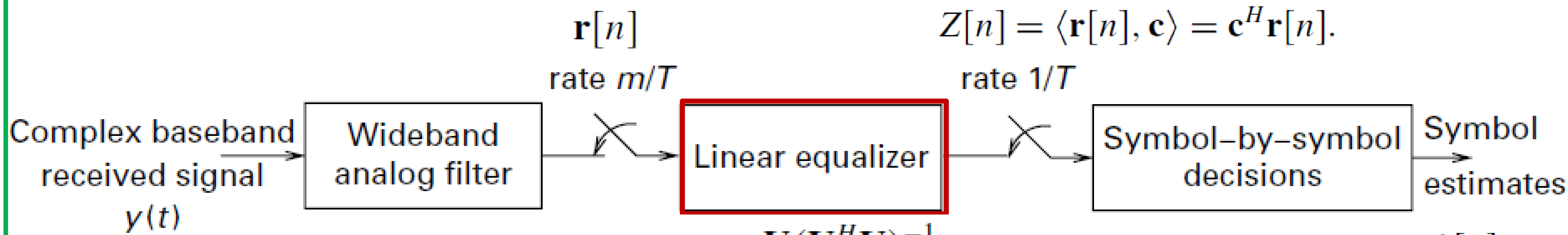
$$\Downarrow$$

$$\boxed{\begin{aligned} \mathbf{c}_{\mathrm{MMSE}} &= \mathbf{R}^{-1}\mathbf{p}, \\ \mathbf{R} = \sigma_b^2 \mathbf{U}\mathbf{U}^H + \mathbf{C}_w = \sigma_b^2 \sum_j \mathbf{u}_j \mathbf{u}_j^H + \mathbf{C}_w, &\quad \mathbf{p} = \sigma_b^2 \mathbf{u}_0 \end{aligned}}$$

$$Z[n] = \mathbf{c}^H \mathbf{r}[n] = b[n](\mathbf{c}^H \mathbf{u}_0) + \sum_{i \neq 0} b[n+i](\mathbf{c}^H \mathbf{u}_i) + \mathbf{c}^H \mathbf{w}[n].$$

$$\Downarrow$$

$$\textbf{SINR} = \frac{\sigma_b^2 |\langle \mathbf{c}, \mathbf{u}_0 \rangle|^2}{\sigma_b^2 \sum_{j \neq 0} |\langle \mathbf{c}, \mathbf{u}_j \rangle|^2 + \mathbf{c}^H \mathbf{C}_w \mathbf{c}}.$$



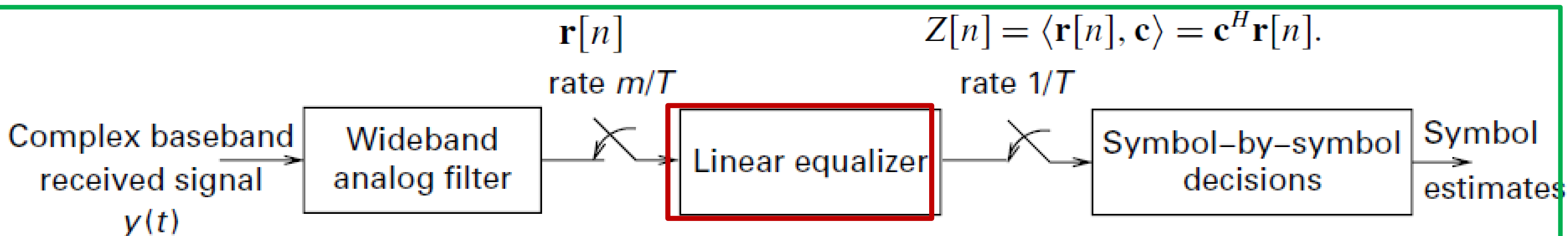
$$\mathbf{c}_{\mathrm{ZF}} = \mathbf{U}(\mathbf{U}^H\mathbf{U})^{-1}\mathbf{e}.$$

$$b[n]$$

$$\mathbf{r}[n] = b[n]\mathbf{u}_0 + \sum_{i \neq 0} b[n+i]\mathbf{u}_i + \mathbf{w}[n]$$

$$Z[n] = \mathbf{c}^H\mathbf{r}[n] = b[n](\mathbf{c}^H\mathbf{u}_0) + \sum_{i \neq 0} \cancel{b[n+i](\mathbf{c}^H\mathbf{u}_i)} + \mathbf{c}^H\mathbf{w}[n].$$

$$\langle \mathbf{c}, \mathbf{u}_0 \rangle = 1$$



$$\mathbf{c}_{\mathrm{MMSE}} = \mathbf{R}^{-1}\mathbf{p},$$

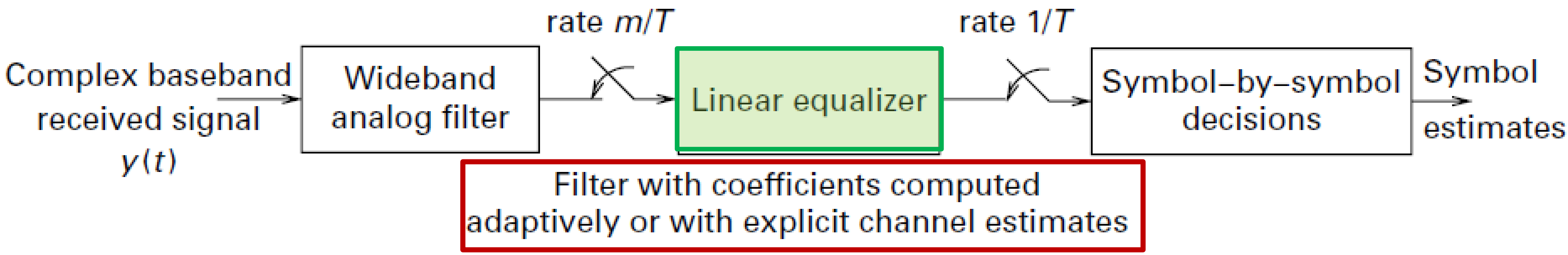
$$b[n]$$

$$\mathbf{r}[n] = b[n]\mathbf{u}_0 + \sum_{i \neq 0} b[n+i]\mathbf{u}_i + \mathbf{w}[n]$$

$$Z[n] = \mathbf{c}^H\mathbf{r}[n] = b[n](\mathbf{c}^H\mathbf{u}_0) + \sum_{i \neq 0} \underbrace{b[n+i](\mathbf{c}^H\mathbf{u}_i) + \mathbf{c}^H\mathbf{w}[n]}.$$

$$\mathbf{SINR} = \frac{\sigma_b^2 |\langle \mathbf{c}, \mathbf{u}_0 \rangle|^2}{\sigma_b^2 \sum_{j \neq 0} |\langle \mathbf{c}, \mathbf{u}_j \rangle|^2 \; + \; \mathbf{c}^H \mathbf{C}_w \mathbf{c}}.$$

Adaptive implementations



Least squares algorithm

$$\mathbf{c}_{LS} = \hat{\mathbf{R}}^{-1}\hat{\mathbf{p}},$$
$$\hat{\mathbf{R}} = \frac{1}{N}\sum_{n=1}^{N}\mathbf{r}[n](\mathbf{r}[n])^H, \qquad \hat{\mathbf{p}} = \frac{1}{N}\sum_{n=1}^{N} b^*[n]\mathbf{r}[n],$$

Recursive least squares algorithm

$$\mathbf{c}[k] = \left(\hat{\mathbf{R}}[k]\right)^{-1}\hat{\mathbf{p}}[k],$$

$$\hat{\mathbf{R}}[k] = \sum_{n=0}^{k}\lambda^{k-n}\mathbf{r}[n](\mathbf{r}[n])^H, \qquad \hat{\mathbf{p}}[k] = \sum_{n=0}^{k}\lambda^{k-n} b^*[n]\mathbf{r}[n].$$

$$\mathbf{c}[k] = \mathbf{c}[k-1] + \frac{e^*[k]\tilde{\mathbf{r}}[k]}{\lambda + (\mathbf{r}[k])^H\tilde{\mathbf{r}}[k]},$$ Matrix inversion Lemma

$$e[k] = b[k] - (\mathbf{c}[k-1])^H\mathbf{r}[k]$$

Least mean squares algorithm

$$\mathbf{c}[k] = \mathbf{c}[k-1] + \mu e^*[k]\mathbf{r}[k]$$ employ gradient descent

normalized LMS (NLMS) algorithm

$$\mathbf{c}[k] = \mathbf{c}[k-1] + \frac{\mu}{P[k]}e^*[k]\mathbf{r}[k],$$
$$P[k] = (\mathbf{r}[k])^H\mathbf{r}[k] + \alpha,$$

4.6.2 Performance analysis

The output of a linear equalizer **c** can be rewritten as

$$Z[n] = \langle \mathbf{r}[n], \mathbf{c} \rangle = \mathbf{c}^H \mathbf{r}[n].$$

$$Z[n] = \mathbf{c}^H \mathbf{r}[n] = b[n](\mathbf{c}^H \mathbf{u}_0) + \sum_{i \neq 0} b[n+i](\mathbf{c}^H \mathbf{u}_i) + \mathbf{c}^H \mathbf{w}[n].$$

$$Z[n] = A_0 b[n] + \boxed{\sum_{i \neq 0} A_i b[n+i]} + \boxed{W[n]}.$$

$A_0 = \langle \mathbf{c}, \mathbf{u}_0 \rangle$ *residual ISI*

$A_i = \langle \mathbf{c}, \mathbf{u}_i \rangle \quad i \neq 0$

zero mean Gaussian noise with variance $v^2 = \sigma^2 ||\mathbf{c}||^2$ per dimension.

If there is no residual ISI as for a ZF equalizer, then error probability computation is straightforward.

However, the residual ISI is nonzero for both MMSE equalization and imperfect ZF equalization. We illustrate the methodology for computing the probability of error in such situations for a real baseband system. Generalizations to complex-valued constellations are straightforward.

Design example--BPSK system

The bit estimate is given by $\hat{b}[n] = \mathrm{sign}(Z[n])$

$$P_e = P[\hat{b}[n] \neq b[n]]. \Longrightarrow P_e = P[Z[n] > 0 | b[n] = +1].$$

For the exact error probability, we condition further on the ISI bits $\mathbf{b}_I = \{b_{n+i}, i \neq 0\}$.

$$\begin{aligned} P_{e|\mathbf{b}_I} &= P[Z[n] > 0 | b[n] = +1, \mathbf{b}_I] \\ &= P[W[n] > -(A_0 + \sum_{i \neq 0} A_i b[n+i])] \\ &= Q\left(\frac{A_0 + \sum_{i \neq 0} A_i b[n+i]}{v}\right). \end{aligned}$$

The complexity is exponential in the number of ISI bits.

We can now average over $\mathbf{b}_I$ to obtain the average error probability:

$$P_e = \mathbb{E}[P_{e|\mathbf{b}_I}].$$

When there are a moderately large number of residual ISI terms, each of which takes small values, an alternative accurate approach is to apply the central limit theorem to approximate the residual ISI as a Gaussian random variable.

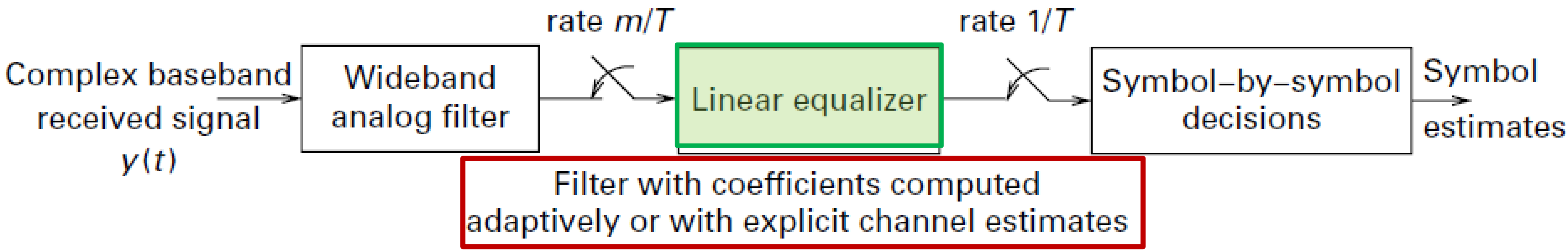
$$v_I^2 = \mathrm{var}\left(\sum_{i\neq 0} A_i b[n+i]\right) = \sum_{i\neq 0} A_i^2 .$$

$$Z[n] = A_0 b[n] + \boxed{\sum_{i\neq 0} A_i b[n+i]} + \boxed{W[n]}, \Longrightarrow Z[n] = A_0 b[n] + N(0, v_I^2 + v^2).$$

The corresponding approximation to the error probability is

$$P_e \approx Q\left(\frac{A_0}{\sqrt{v_I^2 + v^2}}\right) = Q(\sqrt{\mathrm{SIR}}),$$

$$\mathrm{SIR} = \frac{A_0^2}{v_I^2 + v^2} = \frac{|\langle \mathbf{c}, \mathbf{u}_0 \rangle|^2}{\sum_{i\neq 0} \langle \mathbf{c}, \mathbf{u}_i \rangle + \sigma^2 ||\mathbf{c}||^2}.$$



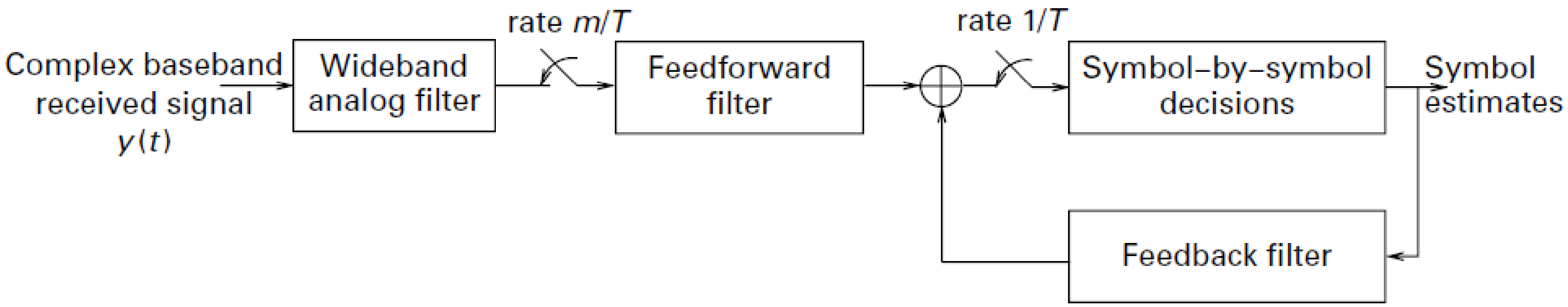
Linear equalizers suppress ISI by projecting the received signal in a direction orthogonal to the interference space: the ZF equalizer does this exactly, the MMSE equalizer does this approximately, taking into account the noise–ISI tradeoff.

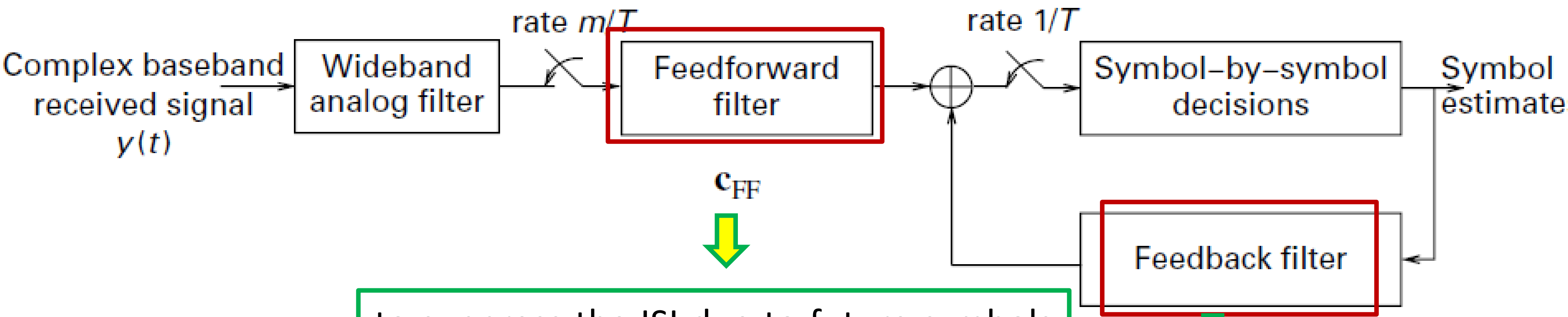
If the desired signal vector component orthogonal to the interference subspace is small, the resulting noise enhancement can be substantial.

4.7 Decision feedback equalization

The DFE alleviates this problem by using feedback from prior decisions to cancel the interference due to the past symbols, and linearly suppressing only the ISI due to future symbols. Since fewer ISI vectors are being suppressed, the noise enhancement is reduced.

The price of this is *error propagation*: an error in a prior decision can cause errors in the current decision via the decision feedback.





to suppress the ISI due to future symbols based on either the ZF or MMSE criteria: ZF-DFE / MMSE-DFE

Cancel the influence of the past symbols

To compute this filter, we simply ignore ISI from the past symbols (assuming that they will be canceled perfectly by decision feedback), and work with the following reduced model including only the ISI from future symbols:

$$\mathbf{r}_n^f = b[n]\mathbf{u}_0 + \sum_{j>0} b[n+j]\mathbf{u}_j + \mathbf{w}[n].$$

The corresponding matrix of signal vectors, containing $\{\mathbf{u}_j, j \geq 0\}$ is denoted by $\mathbf{U}_f$. The ZF and MMSE solutions for $\mathbf{c}_{\mathrm{FF}}$ can be computed by replacing U by $\mathbf{U}_f$:

$$\mathbf{c}_{\mathrm{ZF}} = \mathbf{U}(\mathbf{U}^H\mathbf{U})^{-1}\mathbf{e}.$$

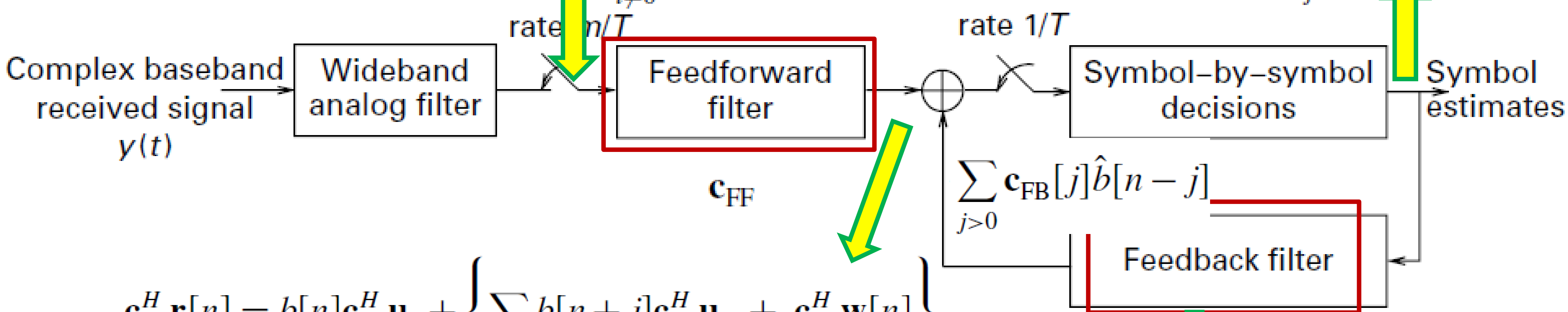
$$\mathbf{c}_{\mathrm{MMSE}} = \mathbf{R}^{-1}\mathbf{p}, \text{ where } \mathbf{R} = \sigma_{\mathrm{b}}^2\mathbf{U}\mathbf{U}^H + \mathbf{C}_{\mathrm{w}} = \sigma_{\mathrm{b}}^2\sum_j \mathbf{u}_j\mathbf{u}_j^H + \mathbf{C}_{\mathrm{w}}, \quad \mathbf{p} = \sigma_{\mathrm{b}}^2\mathbf{u}_0.$$

$$\mathbf{c}_{\mathrm{FF}}^H\mathbf{r}[n] = b[n]\mathbf{c}_{\mathrm{FF}}^H\mathbf{u}_0 + \boxed{\left\{\sum_{j>0} b[n+j]\mathbf{c}_{\mathrm{FF}}^H\mathbf{u}_j \; + \; \mathbf{c}_{\mathrm{FF}}^H\mathbf{w}[n]\right\}} + \boxed{\sum_{j>0} b[n-j]\mathbf{c}_{\mathrm{FF}}^H\mathbf{u}_{-j}}$$

Suppressed ISI of the future symbols

?

$$\mathbf{r}[n] = b[n]\mathbf{u}_0 + \sum_{i \neq 0} b[n+i]\mathbf{u}_i + \mathbf{w}[n], \qquad Z_{\mathrm{DFE}}[n] = \mathbf{c}_{\mathrm{FF}}^H \mathbf{r}[n] + \sum_{j>0} \mathbf{c}_{\mathrm{FB}}[j]\hat{b}[n-j].$$



$$\mathbf{c}_{\mathrm{FF}}^H \mathbf{r}[n] = b[n]\mathbf{c}_{\mathrm{FF}}^H \mathbf{u}_0 + \left\{ \sum_{j>0} b[n+j]\mathbf{c}_{\mathrm{FF}}^H \mathbf{u}_j \; + \; \mathbf{c}_{\mathrm{FF}}^H \mathbf{w}[n] \right\}$$

$$+ \boxed{\sum_{j>0} b[n-j]\mathbf{c}_{\mathrm{FF}}^H \mathbf{u}_{-j}.}$$

Cancel the influence of the past symbols

$$\mathbf{c}_{\mathrm{FB}}[j] = -\mathbf{c}_{\mathrm{FF}}^H \mathbf{u}_{-j}, \; j > 0,$$

$$Z_{\mathrm{DFE}}[n] \; = \; b[n]\mathbf{c}_{\mathrm{FF}}^H \mathbf{u}_0 + \left\{ \sum_{j>0} b[n+j]\mathbf{c}_{\mathrm{FF}}^H \mathbf{u}_j \; + \; \mathbf{c}_{\mathrm{FF}}^H \mathbf{w}[n] \right\}$$

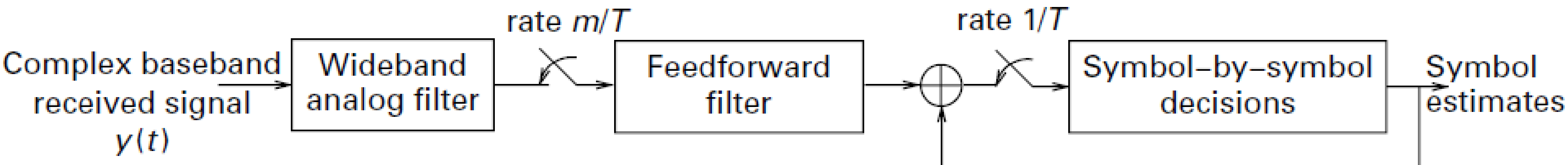
$$+ \sum_{j>0} (b[n-j] - \hat{b}[n-j])\mathbf{c}_{\mathrm{FF}}^H \mathbf{u}_{-j},$$

Setting $\mathbf{U}_\mathrm{p}$ as the matrix with the past ISI vectors $\{\mathbf{u}_{-1}, \mathbf{u}_{-2}, \ldots\}$ as columns

$$\mathbf{c}_\mathrm{FB}[j] = -\mathbf{c}_\mathrm{FF}^H \mathbf{u}_{-j},\ j > 0, \Rightarrow \quad \mathbf{c}_\mathrm{FB} = -\mathbf{c}_\mathrm{FF}^H \mathbf{U}_\mathrm{p},$$

$$\mathbf{c}_\mathrm{FB} = (\mathbf{c}_\mathrm{FB}[K_\mathrm{p}], \ldots, \mathbf{c}_\mathrm{FB}[1])^T$$

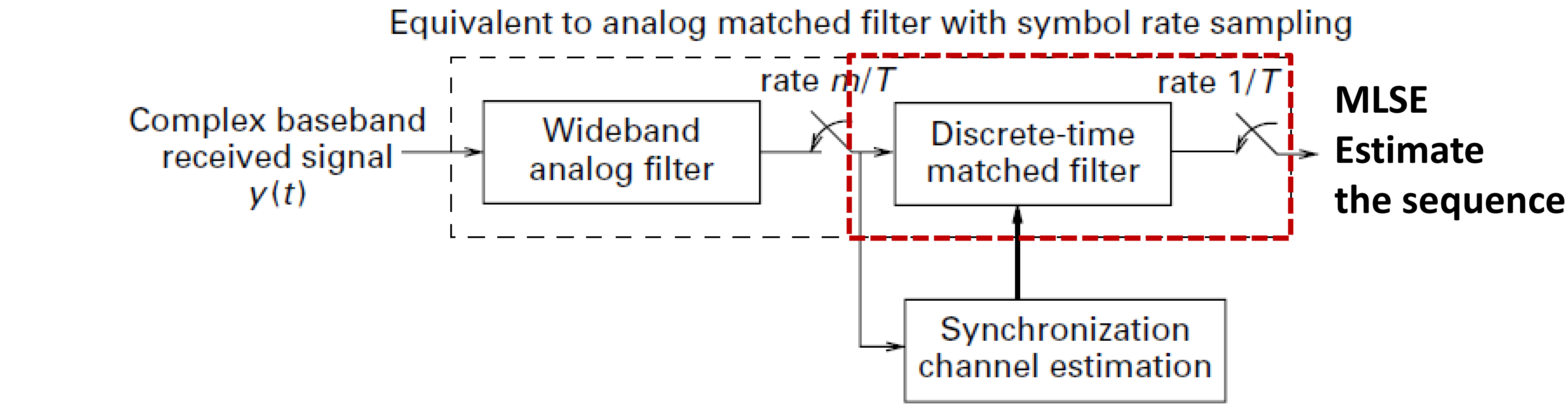
the number of past symbols being fed back.

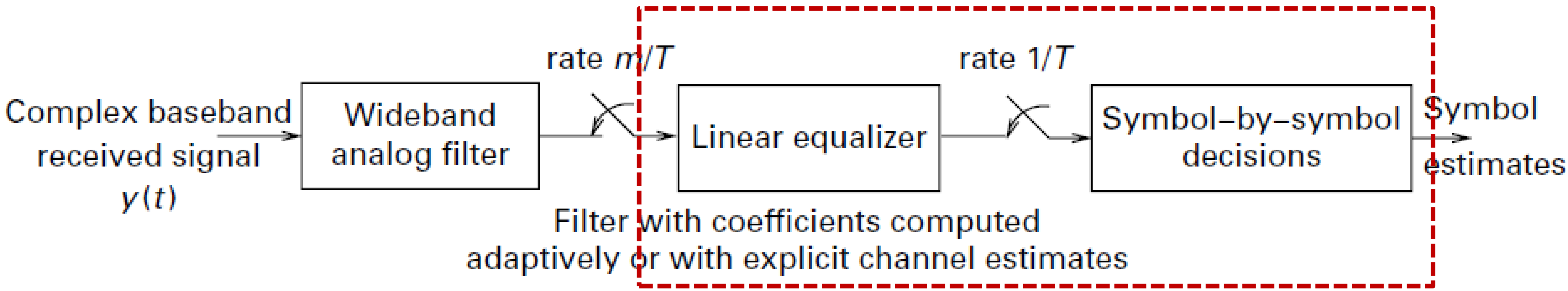


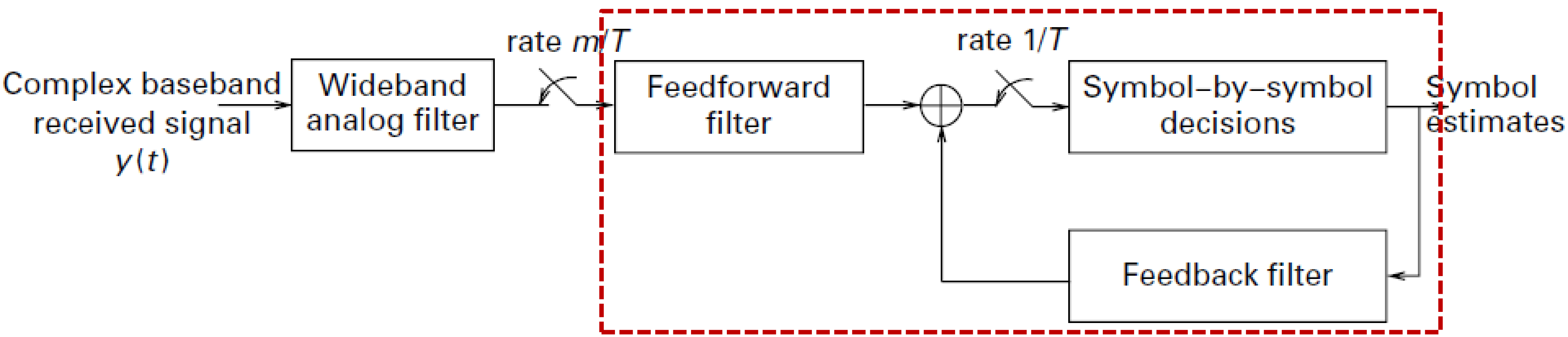
$$\mathbf{c}_\mathrm{FF} = \mathbf{U}_f \left(\mathbf{U}_f^H \mathbf{U}_f\right)^{-1} \mathbf{e}.$$

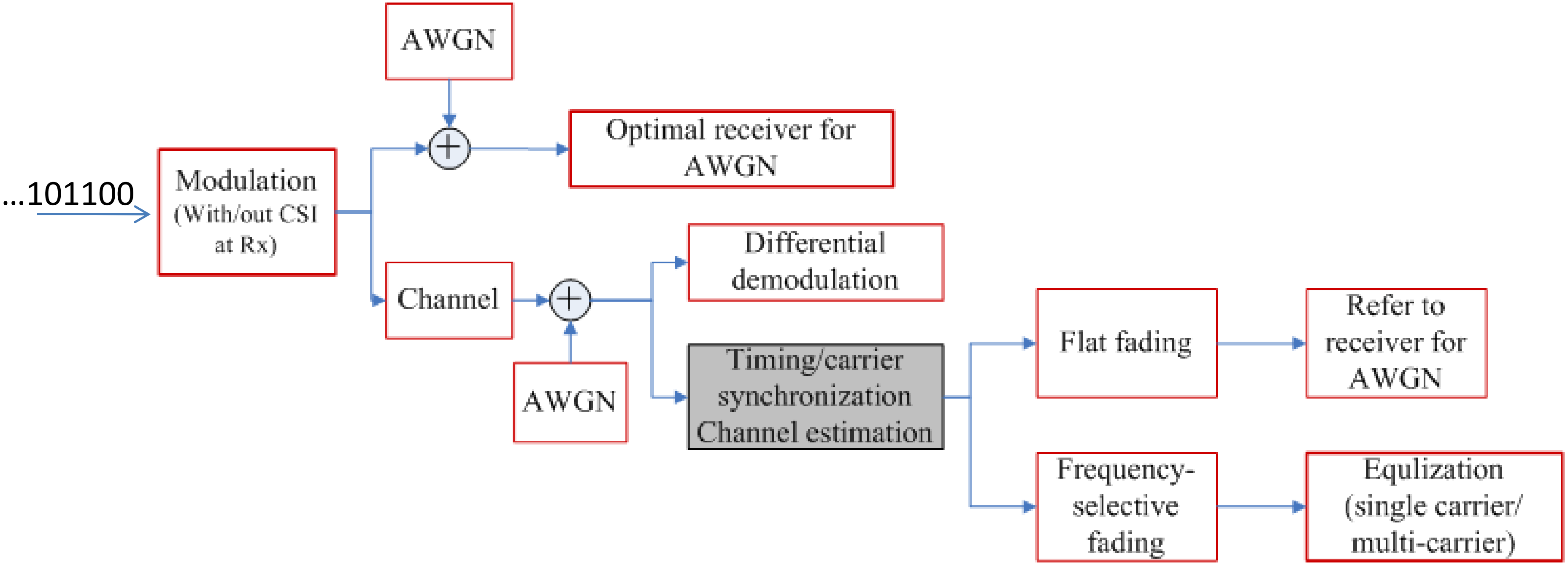
$$\mathrm{c}_{FF} = \mathrm{R}^{-1}\mathrm{P}, \mathrm{R} = \sigma_b^2 \mathrm{U}_f \mathrm{U}_f^H + \mathrm{C}_w, \mathrm{P} = \sigma_b^2 \mathrm{u}_0$$

$$\mathbf{c}_\mathrm{FB} = -\mathbf{c}_\mathrm{FF}^H \mathbf{U}_\mathrm{p},$$


Equivalent to analog matched filter with symbol rate sampling
Complex baseband received signal y(t)
Wideband analog filter
rate m/T
Discrete-time matched filter
rate 1/T
MLSE
Estimate the sequence
Synchronization channel estimation


Complex baseband received signal y(t)
Wideband analog filter
rate m/T
Linear equalizer
rate 1/T
Symbol–by–symbol decisions
Symbol estimates
Filter with coefficients computed adaptively or with explicit channel estimates


Complex baseband received signal y(t)
Wideband analog filter
rate m/T
Feedforward filter
rate 1/T
Symbol–by–symbol decisions
Symbol estimates
Feedback filter


...101100
Modulation
(With/out CSI
at Rx)
AWGN
Optimal receiver for
AWGN
Channel
Differential
demodulation
AWGN
Timing/carrier
synchronization
Channel estimation
Flat fading
Refer to
receiver for
AWGN
Frequency-
selective
fading
Equlization
(single carrier/
multi-carrier)