Chpater 8 Digital Transmission through Bandlimited AWGN Channels

Text. [1] J. G. Proakis and M. Salehi, Communication Systems Engineering, 2/e. Prentice Hall, 2002.

Digital communication over a channel is modeled as a linear filter with a bandwidth limitation.

Bandlimited channels most frequently are encountered in telephone channels, microwave LOS radio channels, satellite channels, and underwater acoustic channels.

The transmitted signals must be designed to satisfy the bandwidth constraint imposed by the channel, that is, the transmitted signals must be shaped to restrict their bandwidth to that available on the channel.

8.1 Digital Transmission through Bandlimited Channels

A bandlimited channel is characterized as a linear filter with impulse response $c(t)$ and frequency response $C(f)$ which is given by

$$C(f) = \int_{-\infty}^{\infty} c(t) e^{-j2\pi f t} dt . \tag{8.1.1}$$

If the channel is a baseband channel that is bandlimited to B_c Hz, then

$C(f) = 0$ for $|f| > B_c$.

Any frequency components with frequency higher than B_c Hz will not be passed by the channel which is bandlimited to $W = B_c$ Hz as shown in Figure 8.1.

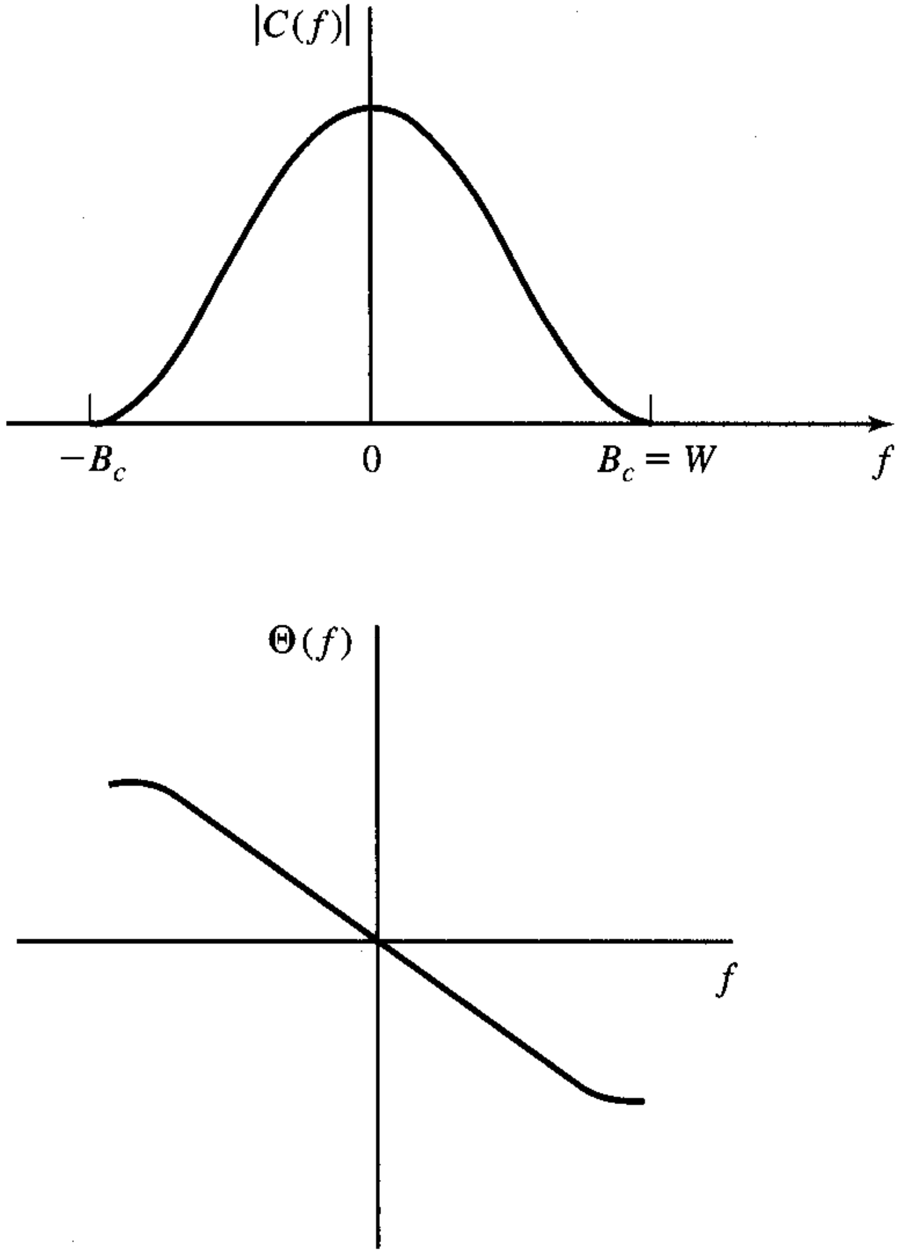


Figure 8.1 Magnitude and phase responses of bandlimited channel.

Let W denote the bandwidth limitation of the signal and the channel.

Suppose that the input to a bandlimited channel is a signal $g_T(t)$.

Then, the output of the channel corresponding to the input $g_T(t)$ is given by

$$h(t) = \int_{-\infty}^{\infty} c(\tau) g_T(t-\tau) d\tau$$

$$= c(t) * g_T(t) \tag{8.1.2}$$

or, in the frequency domain, we have

$$H(f) = C(f) G_T(f) \tag{8.1.3}$$

where $G_T(f)$ is the spectrum (Fourier transform) of the signal $g_T(t)$ and $H(f)$ is the spectrum of $h(t)$.

Assume that the signal at the input to the demodulator (that is, at the output of the channel) is corrupted by AWGN which is given by $h(t) + n(t)$ where $n(t)$ is the AWGN.

In the presence of AWGN, a demodulator having a filter which is matched to the signal $h(t)$ maximizes the SNR at its output.

Let the received signal $h(t)+n(t)$ is passed through the matched filter of which frequency response is given by

$$G_R(f) = H^*(f)e^{-j2\pi f t_0} \tag{8.1.4}$$

where t_0 is time delay at which the filter output is sampled.

The signal component at the output of the matched filter at the sampling instant $t=t_0$ is given by

$$\begin{aligned} y_S(t_0) &= \int_{-\infty}^{\infty} |H(f)|^2 df \\ &= \varepsilon_h \end{aligned} \tag{8.1.5}$$

which is the energy in the channel output $h(t)$.

The noise component at the output of the matched filter has zero mean and a power-spectral density given by

$$S_n(f) = \frac{N_0}{2}|H(f)|^2. \tag{8.1.6}$$

Hence, the noise power at the output of the matched filter has a variance

$$\sigma_n^2 = \int_{-\infty}^{\infty} S_n(f)df = \frac{N_0}{2}\int_{-\infty}^{\infty} |H(f)|^2 df = \frac{N_0 \varepsilon_h}{2} \tag{8.1.7}$$

The SNR at the output of the matched filter is given by

$$\left(\frac{S}{N}\right)_0 = \frac{\varepsilon_h^2}{\frac{N_0 \varepsilon_h}{2}} = \frac{2\varepsilon_h}{N_0} \tag{8.1.8}$$

which is the same as SNR at the output of the matched filter in Chapter 7 except that the received signal nergy ε_h has replaced the transmitted signal energy ε_S.

Note that the filter impulse response is matched to the signal component $h(t)$ in the received signal instead of the transmitted signal.

Note that to implement the matched filter at the receiver, $h(t)$ (or, equivalently, the channel impulse response $c(t)$) must be known to the receiver.

Ex. 8.1.1

The signal $g_T(t)$ is given by

$$g_T(t) = \frac{1}{2}\left[1 + \cos\frac{2\pi}{T}\left(t - \frac{T}{2}\right)\right], \qquad 0 \le t \le T,$$

which is shown in Figure 8.2(b).

$g_T(t)$ is transmitted through a baseband channel with frequency-response characteristic as shown in Figure 8.2(a).

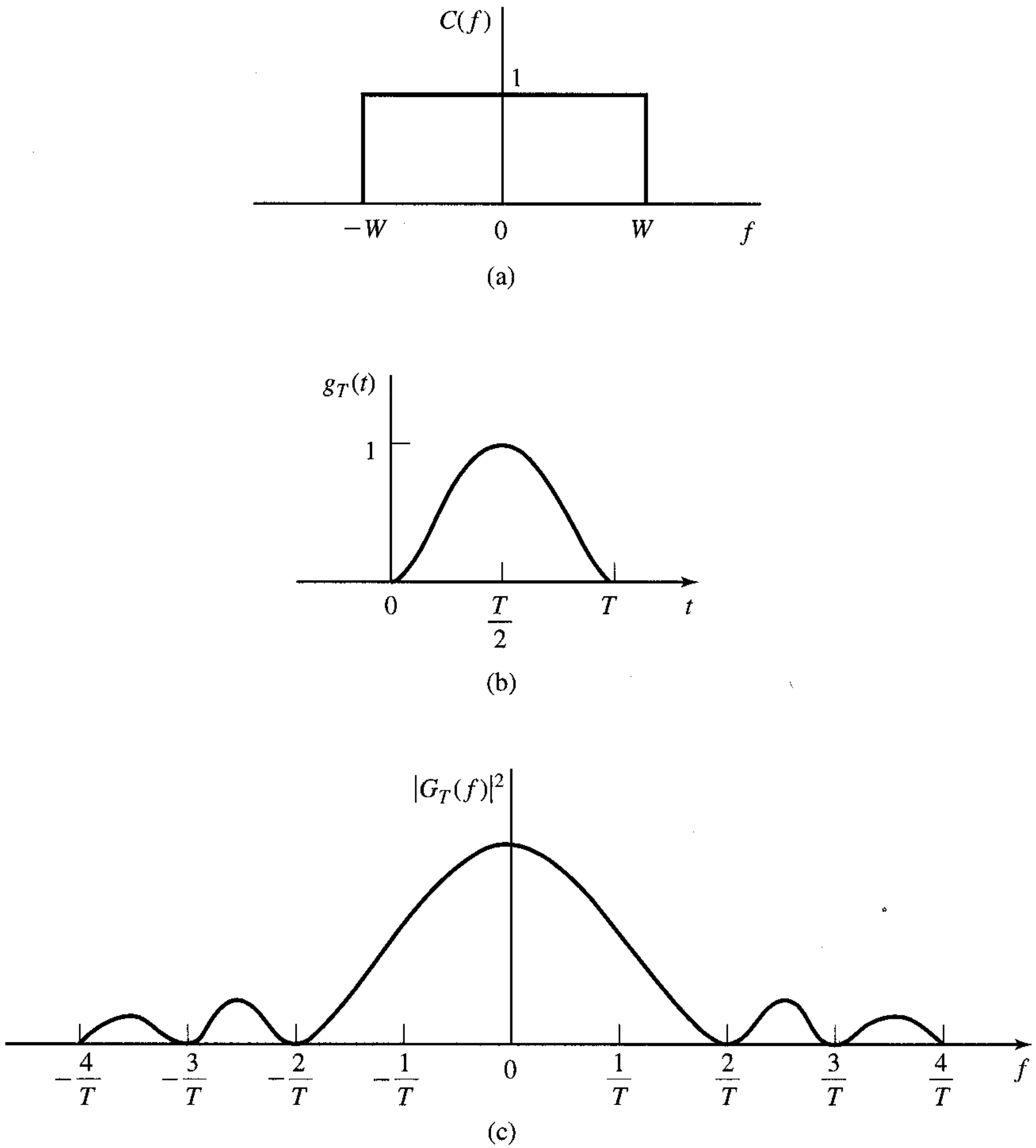


Figure 8.2 The signal pulse in (b) is transmitted through the ideal bandlimited channel shown in (a). The spectrum of $g_T(t)$ is shown in (c).

The channel output is corrupted by AWGN with power-spectral density $\frac{N_0}{2}$.

Determine the matched filter to the received signal and the output SNR.

Solution

The spectrum of the signal is given by

$$G_T(f) = \frac{T}{2}\frac{\sin \pi fT}{\pi fT(1-f^2T^2)}e^{-j\pi f T}$$

$$= \frac{T}{2}\frac{\operatorname{sinc} \pi fT}{(1-f^2T^2)}e^{-j\pi f T}$$

of which square, $|G_T(f)|^2$, is shown in Figure 8.2(c).

Hence,

$$H(f) = C(f)G_T(f)$$

$$= \begin{cases} G_T(f), & |f| \le W, \\ 0, & \text{otherwise.} \end{cases}$$

Then, the signal component at the output of the filter matched to $H(f)$ is given by

$$
\begin{aligned}
\varepsilon_h &= \int_{-\infty}^{\infty} |H(f)|^2 df \\
&= \int_{-W}^{W} |G_T(f)|^2 \, df \\
&= \frac{1}{(2\pi)^2} \int_{-W}^{W} \frac{(\sin \pi fT)^2}{f^2(1-f^2T^2)^2} \, df \\
&= \frac{T}{(2\pi)^2} \int_{-WT}^{WT} \frac{\sin^2 \pi\alpha}{\alpha^2(1-\alpha^2)^2} \, d\alpha .
\end{aligned}
$$

The variance of the noise component is given by

$$
\begin{aligned}
\sigma_n^2 &= \frac{N_0}{2} \int_{-W}^{W} |G_T(f)|^2 df \\
&= \frac{N_0 \varepsilon_h}{2} .
\end{aligned}
$$

Hence, the output SNR is given by

$$
\left(\frac{S}{N}\right)_0 = \frac{2\varepsilon_h}{N_0} .
$$

In this example, only a part of the transmitted signal energy is received, since the signal at the input to the channel is not bandlimited.

The amount of signal energy at the output of the matched filter depends on the channel bandwidth W when the signal pulse duration is fixed (see Problem 8.1).

The maximum value of ε_h is obtained by letting $W \to \infty$, that is,

$$\begin{aligned}\max \varepsilon_h &= \lim_{W\to\infty} \int_{-W}^{W} |G_T(f)|^2 \, df \\ &= \int_{-\infty}^{\infty} |G_T(f)|^2 \, df \\ &= \int_0^T g_T^2(t)dt \, .\end{aligned}$$

Note that the performance of the system is determined by ε_h, the energy in the received signal $h(t)$.

To maximize the received SNR, we have to make sure that the power-spectral density of the transmitted signal matches the frequency band of the channel.

8.1.1 Digital PAM Transmission through Bandlimited Baseband Channels

Consider the baseband PAM communication system shown by the block diagram in Figure 8.3.

The system consists of a transmitting filter having an impulse response $g_T(t)$, the linear filter channel with AWGN, a receiving filter with impulse response $g_R(t)$, a sampler that periodically samples the output of receiving filter, and a symbol detector.

The sampler needs a timing signal whch is extracted from the received signal (see Section 7.8) to serve as a clock to specify the appropriate time instants to sample the output of the receiving filter.

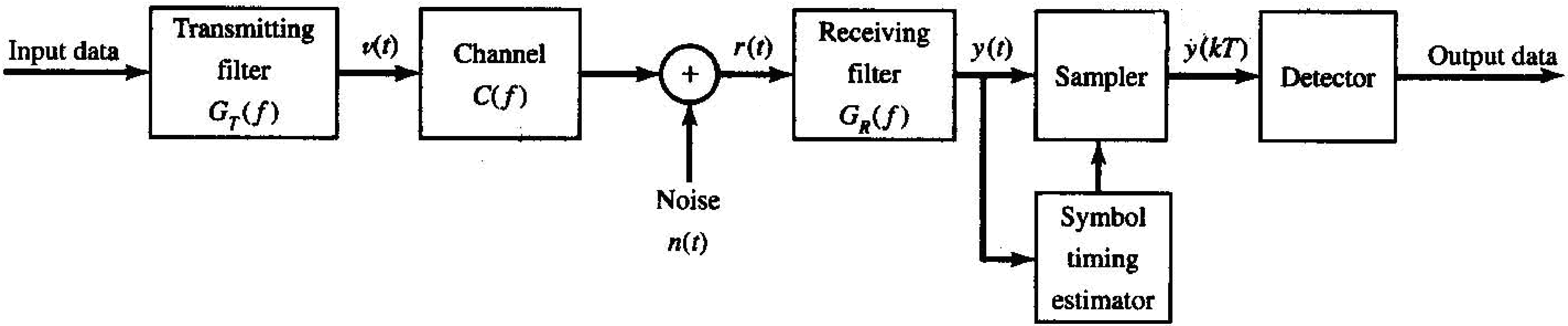


Figure 8.3 Block diagram of digital PAM system.

The input binary data sequence of bit rate R_b from the source is subdivided into k-bit symbols and each 2^k-ary symbol a_n is mapped into a corresponding amplitude level which modulates the output of the transmitting filter.

The baseband signal at the output of the transmitting filter (or at the input to the channel) is given by

$$v(t) = \sum_{n=-\infty}^{\infty} a_n g_T(t - nT) \tag{8.1.9}$$

where $T=\dfrac{k}{R_b}$ is the symbol interval.

Note that the symbol rate is given by $\dfrac{1}{T}=\dfrac{R_b}{k}$.

The received signal at the demodulator (or the channel output) is given by

$$r(t)=\sum_{n=-\infty}^{\infty} a_n h(t-nT)+n(t) \tag{8.1.10}$$

where $h(t)$ is the impulse response of the cascade of the transmitting filter and the channel, that is, $h(t)=c(t)*g_T(t)$,

$c(t)$ is the impulse response of the channel, and

$n(t)$ is an AWGN.

The received signal is passed through a linear receiving filter with impulse response $g_R(t)$ and frequency response $G_R(f)$.

If $g_R(t)$ is matched to $h(t)$, then its output SNR becomes a maximum at the proper sampling instant.

The output of the receiving filter is given by

$$y(t) = \sum_{n=-\infty}^{\infty} a_n x(t - nT) + v(t) \tag{8.1.11}$$

where $x(t) = h(t) * g_R(t) = g_T(t) * c(t) * g_R(t)$ and

$v(t) = n(t) * g_R(t)$ is the additive noise at the output of the receiving filter.

To recover the information symbols $\{a_n\}$, the output of the receiving filter is sample periodically with the interval of T seconds.

The output of the sampler is given by

$$y(mT) = \sum_{n=-\infty}^{\infty} a_n x(mT - nT) + v(mT) \tag{8.1.12}$$

or, equivalently,

$$y_m = \sum_{n=-\infty}^{\infty} a_n x_{m-n} + v_m$$

$$= x_0 a_m + \sum_{\substack{n=-\infty \\ n\neq m}}^{\infty} a_n x_{m-n} + v_m \qquad (8.1.13)$$

where $x_m = x(mT)$ and

$v_m = v(mT)$ for $m = \cdots, -2, -1, 0, 1, 2, \cdots.$

The first term, $x_0 a_m$, on the right-hand side (RHS) of (8.1.13) is the desired symbol a_m scaled by the gain parameter x_0.

When the receiving filter is matched to the received signal $h(t)$, the scale factor is given by

$$x_0 = \int_{-\infty}^{\infty} h^2(t)dt$$

$$= \int_{-\infty}^{\infty} | H(f)|^2 df$$

$$= \int_{-W}^{W} | G_T(f)|^2 | C(f)|^2 \, df$$

$$= \varepsilon_h \qquad (8.1.14)$$

as shown in (8.1.4) and (8.1.5).

The second term on the RHS of (8.1.13) represents the **intersymbol interference (ISI)** which is the effect of the other symbols to the desired symbols at the sampling instant $t = mT$.

In general, ISI degrades the performance of the digital communic*a*tion system.

The third term, v_m, on the RHS of (8.1.13) is the additive noise and is a zero-mean Gaussian random variable with variance $\sigma_v^2 = \frac{N_0 \varepsilon_h}{2}$ as given by (8.1.7).

By appropriate design of the transmitting and receiving filters, it is possible to satisfy the condition $x_n = 0$ for all $n \neq 0$, so that the ISI term vanishes.

In this case, the only term which causes errors in the received digital sequence is the additive noise.

8.1.2 Digital Transmission through Bandlimited Bandpass Channels

The development given in Section 8.1.1 for baseband PAM is easily extended to carrier modulation via PAM, and QAM and PSK.

In a carrier-amplitude modulated signal (or bandpass PAM or ASK), the baseband PAM given by $v(t)$ in (8.1.9) modulates the carrier, so that the transmitted signal is given by

$$u(t) = v(t)\cos 2\pi f_c t, \tag{8.1.15}$$

which implies that the baseband signal $v(t)$ is shifted in frequency by f_c.

A QAM signal is a bandpass signal.

A rectangular QAM signal may be viewed as two amplitude-modulated carrier signals in phase quadrature.

That is, the QAM signal is given by

$$u(t) = v_c(t)\cos 2\pi f_c t + v_s(t)\sin 2\pi f_c t \tag{8.1.16}$$

where

$$v_c(t) = \sum_{n=-\infty}^{\infty} a_{nc} g_T(t - nT),$$

$$v_s(t) = \sum_{n=-\infty}^{\infty} a_{ns} g_T(t - nT) \tag{8.1.17}$$

and $\{a_{nc}\}$ and $\{a_{ns}\}$ are the two sequences of amplitudes carried on the two quadrature carriers.

An equivalent complex-valued baseband signal of the QAM signal is given by

$$v(t) = v_c(t) - jv_s(t)$$

$$= \sum_{n=-\infty}^{\infty} (a_{nc} - ja_{ns}) g_T(t - nT)$$

$$= \sum_{n=-\infty}^{\infty} a_n g_T(t - nT) \tag{8.1.18}$$

where $a_n = a_{nc} - ja_{ns}$ and the sequence $\{a_n\}$ is a complex-valued sequence representing the signal points in the QAM signal constellation.

From (8.1.16) and (8.1.18), the corresponding bandpass QAM signal is given by

$$u(t) = \mathrm{Re}\left[v(t)e^{j2\pi f_c t}\right]. \tag{8.1.19}$$

Similarly to (8.1.19), an equivalent complex-valued baseband (or PSK) signal is given by

$$v(t) = \sum_{n=-\infty}^{\infty} a_n g_T(t-nT) \tag{8.1.20}$$

and the sequence $\{a_n\}$ takes the value from the set of possible (phase) values $\{e^{-j2\pi\frac{m}{M}},\ m = 0, 1, \cdots, M-1\}$.

All three carrier-modulated signals for PAM, QAM, and PSK can be represented as in (8.1.19) and (8.1.20), where the only difference is in the values of the transmitted sequence $\{a_n\}$.

The signal $v(t)$ given by (8.1.19) or (8.1.20) is called the **equivalent lowpass signal**.

In the case of QAM and PSK, the equivalent lowpass signal $v(t)$ is a complex-valued baseband signal because the information-bearing sequence $\{a_n\}$ is complex-valued.

In the case of PAM, the equivalent lowpass signal $v(t)$ is a real-valued baseband signal.

Suppose that the channel has the impulse response of the equivalent lowpass channel $c(t)$.

Then, when transmitted through the bandpass channel, the received bandpass signal is given by

$$w(t) = \mathrm{Re}\left[r(t) e^{j2\pi f_c t} \right] \quad (8.1.21)$$

where $r(t)$ is the equivalent lowpass (baseband) signal, which is given by

$$r(t) = \sum_{n=-\infty}^{\infty} a_n h(t - nT) + n(t) \quad (8.1.22)$$

where $h(t)$ is the impulse response of the cascade of the transmitting filter and the channel, that is, $h(t) = c(t) * g_T(t)$, and

$n(t)$ is the additive Gaussian noise expressed as an equivalent lowpass (baseband) noise.

The received bandpass signal can be converted to a baseband signal by multiplying $w(t)$ with the quadrature carrier signals $\cos 2\pi f_c t$ and $\sin 2\pi f_c t$ and eliminating the double frequency terms by passing the two quadrature components through two separate lowpass filters, as shown in Figure 8.4.

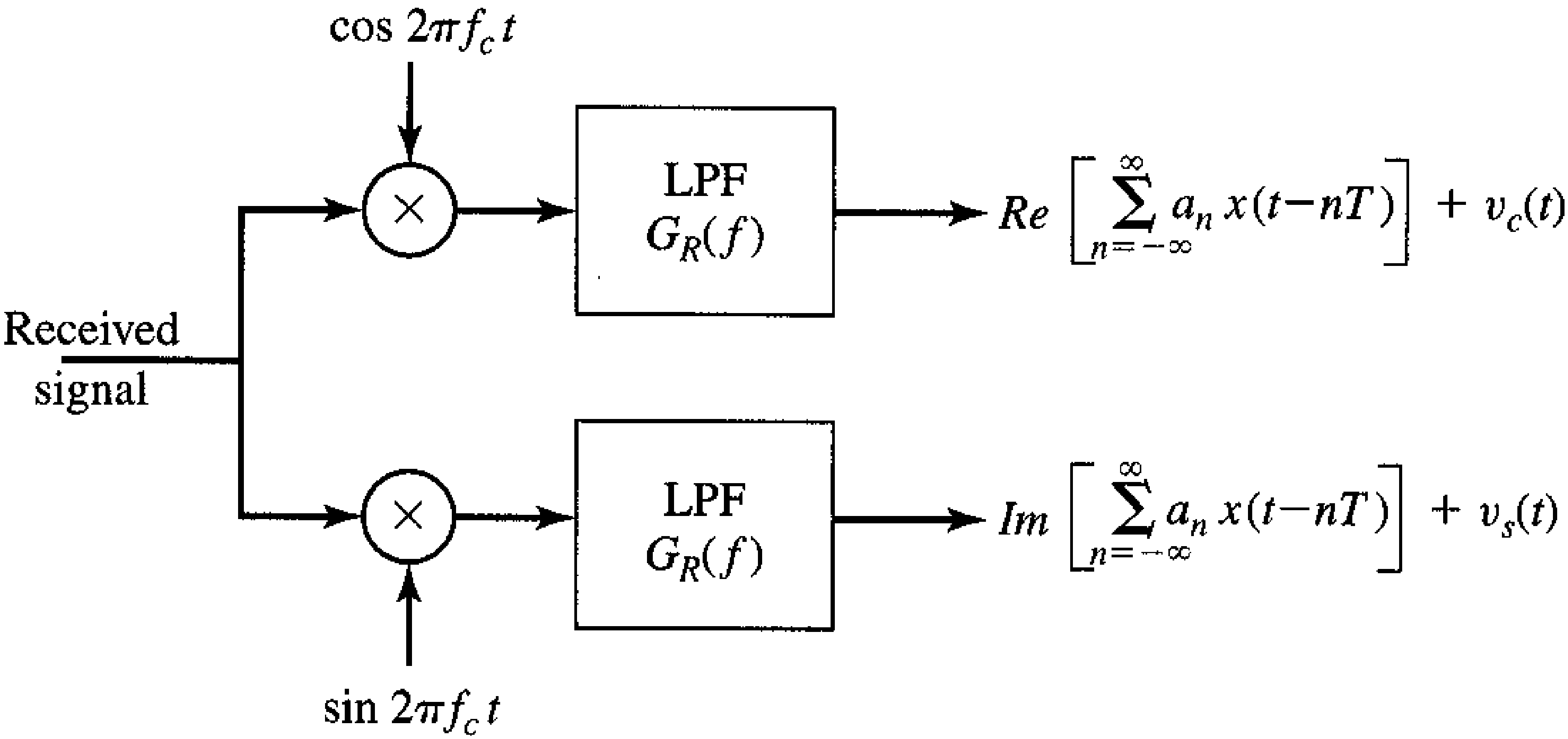


Figure 8.4 Conversion of the bandpass received signal to baseband.

Assume each of the lowpass filters in Figure 8.4 has an impulse response $g_R(t)$.

Then, we can represent the two quadrature components at the outputs of the two lowpass filters as an equivalent complex-valued signal

$$y(t) = \sum_{n=-\infty}^{\infty} a_n x(t - nT) + v(t) \tag{8.1.23}$$

which is identical to (8.1.11) for the real baseband signal.

Hence, the signal design problem for bandpass signals is basically the same as that for baseband signals.

8.2 Power Spectral Density of Digitally Modulated Signals

8.2.1 Power Spectral Density of the Baseband Signal

The equivalent baseband transmitted signal for a digital PAM, PSK, or QAM signal is represented in the general form as

$$v(t) = \sum_{n=-\infty}^{\infty} a_n g_T(t - nT) \tag{8.2.1}$$

where $\{a_n\}$ is the sequence of values selected from either a PAM, QAM, or PSK signal constellation corresponding to the information symbols from the source, and

$g_T(t)$ is the impulse response of the transmitting filter.

Since the information sequence $\{a_n\}$ is random, $v(t)$ is a sample function of a random process $V(t)$.

The mean function of the random variable (or the baseband transmitted signal) $v(t)$ is given by

$$E[V(t)] = \sum_{n=-\infty}^{\infty} E[a_n] g_T(t - nT)$$

$$= m_a \sum_{n=-\infty}^{\infty} g_T(t-nT) \tag{8.2.2}$$

where m_a is the mean of the random variable a_n.

Note that, since although m_a is a constant, the term $\sum_n g_t(t-nT)$ in (8.2.2) is a periodic function with period T, the mean function of $V(t)$ is periodic with period T.

The autocorrelation function of $V(t)$ is given by

$$R_V(t+\tau, t) = E[V^*(t)V(t+\tau)]$$

$$= \sum_{n=-\infty}^{\infty} \sum_{m=-\infty}^{\infty} E[a_n^* \, a_m] g_T(t-nT) g_T(t+\tau-mT)\,. \tag{8.2.3}$$

Assume that the information sequence $\{a_n\}$ is wide-sense stationary with the autocorrelation function given by

$$R_a(n) = E[a_m^* a_{n+m}]\,. \tag{8.2.4}$$

Then, (8.2.3) becomes

$$R_V(t+\tau,t) = \sum_{n=-\infty}^{\infty}\sum_{m=-\infty}^{\infty} R_a(m-n) g_T(t-nT) g_T(t+\tau-mT)$$

$$= \sum_{m=-\infty}^{\infty} R_a(m) \sum_{n=-\infty}^{\infty} g_T(t-nT) g_T(t+\tau-nT-mT)\,. \quad (8.2.5)$$

where the second summation $\sum_{n=-\infty}^{\infty} g_T(t-nT) g_T(t+\tau-nT-mT)$ is periodic with period T.

Consequently, the autocorrelation function $R_V(t+\tau,t)$ is periodic in the variable t, that is,

$$R_V(t+T+\tau, t+T) = R_V(t+\tau,t)\,. \quad (8.2.7)$$

Since the random process $V(t)$ has a periodic mean function and a periodic autocorrelation function, the random process $V(t)$ is **cyclostationary** (see Definition 4.2.7).

The power-spectral density of a cyclostationary process $V(t)$ is determined by averaging the autocorrelation function $R_V(t+\tau,t)$ over a period T and then computing the Fourier transform of the average autocorrelation function (see Corollary to Theorem 4.3.1).

The average autocorrelation function of $V(t)$ is given by

$$\begin{aligned}
\bar{R}_V(\tau) &= \frac{1}{T}\int_{-\frac{T}{2}}^{\frac{T}{2}} R_V(t+\tau,t)dt \\
&= \sum_{m=-\infty}^{\infty} R_a(m) \sum_{n=-\infty}^{\infty} \frac{1}{T}\int_{-\frac{T}{2}}^{\frac{T}{2}} g_T(t-nT)g_T(t+\tau-nT-mT)dt \\
&= \sum_{m=-\infty}^{\infty} R_a(m) \sum_{n=-\infty}^{\infty} \frac{1}{T}\int_{nT-\frac{T}{2}}^{nT+\frac{T}{2}} g_T(t)g_T(t+\tau-mT)dt \\
&= \frac{1}{T}\sum_{m=-\infty}^{\infty} R_a(m)\int_{-\infty}^{\infty} g_T(t)g_T(t+\tau-mT)dt
\end{aligned} \tag{8.2.8}$$

The integral in (8.2.8) is the time-autocorrelation function of $g_T(t)$ which is defined as (See (2.3.1))

$$R_g(\tau) \triangleq \int_{-\infty}^{\infty} g_T(t)g_T(t+\tau)dt\,. \tag{8.2.9}$$

From (8.2.8) and (8.2.9) it becomes

$$\bar{R}_V(\tau) = \frac{1}{T}\sum_{m=-\infty}^{\infty} R_a(m) R_g(\tau - mT) \tag{8.2.10}$$

which has the form of a convolution sum.

Hence the Fourier transform of the average autocorrelation function of $V(t)$ in (8.2.10) is given by

$$\begin{aligned} S_V(f) &= \int_{-\infty}^{\infty} \bar{R}_V(\tau) e^{-j2\pi f\tau} d\tau \\ &= \frac{1}{T}\sum_{m=-\infty}^{\infty} R_a(m) \int_{-\infty}^{\infty} R_g(\tau - mT) e^{-j2\pi f\tau} d\tau \\ &= \frac{1}{T} S_a(f) |G_T(f)|^2 \end{aligned} \tag{8.2.11}$$

where $S_a(f)$ is the power spectral density of the information sequence $\{a_n\}$ given by

$$S_a(f) = \sum_{m=-\infty}^{\infty} R_a(m) e^{-2\pi f\, mT} \tag{8.2.12}$$

and $G_T(f)$ is the transfer function of the transmitting filter.

$|G_T(f)|^2$ is the Fourier transform of $R_g(\tau)$.

From (8.2.11), notice that the power-spectral density $S_V(f)$ of the transmitted signal $V(t)$ depends on (1) the transfer function $G_T(f)$ of the transmitting filter and (2) the power spectral density $S_a(f)$ of the information sequence $\{a_n\}$.

Both $G_T(f)$ and $S_a(f)$ can be designed to adjuct the shape of the power spectral density of the transmitted signal.

Examine the dependence of $S_V(f)$ on $S_a(f)$.

First, we observe that, for an arbitrary autocorrelation function $R_a(m)$, the corresponding power-spectral density $S_a(f)$ is periodic in frequency with period $\frac{1}{T}$.

Note that $S_a(f)$ in (8.2.12) is an exponential Fourier series with the Fourier coefficients $\{R_a(m)\}$.

Consequently, the autocorrelation function of the information sequence $\{a_n\}$ is given by

$$R_a(m) = T\int_{-\frac{1}{2T}}^{\frac{1}{2T}} S_a(f)e^{j2\pi f mT}df \,. \tag{8.2.13}$$

Second, suppose that the information symbols in the sequence $\{a_n\}$ are mutually uncorrelated.

Then, the autocorrelation function of the information sequence $\{a_n\}$ is given by

$$R_a(m) = \begin{cases} \sigma_a^2 + m_a^2, & m = 0, \\ m_a^2, & m \neq 0, \end{cases} \tag{8.2.14}$$

where $\sigma_a^2 = E[a_n^2] - m_a^2$ is the variance of an information symbol.

From (8.2.12) and (8.2.14), the power-spectral density of the information sequence $\{a_n\}$ is given by

$$S_a(f) = \sigma_a^2 + m_a^2 \sum_{m=-\infty}^{\infty} e^{-j2\pi f mT} \,. \tag{8.2.15}$$

The second term on the RHS of (8.2.15) is periodic with period $\dfrac{1}{T}$ and can be viewed as the Fourier series of a periodic sequence of impulses each of which has an area $\dfrac{1}{T}$ (see Table 2.1).

Therefore, (8.2.15) is expressed as

$$S_a(f) = \sigma_a^2 + m_a^2 \frac{1}{T} \sum_{m=-\infty}^{\infty} \delta(f - \frac{m}{T}) . \tag{8.2.16}$$

From (8.2.11) and (8.2.16), the power-spectral density of the transmitted signal $V(t)$, when the sequence of information symbols is uncorrelated, is given by

$$S_V(f) = \frac{\sigma_a^2}{T} |G_T(f)|^2 + \frac{m_a^2}{T^2} \sum_{m=-\infty}^{\infty} \left| G_T\left(\frac{m}{T}\right)\right|^2 \delta\left(f - \frac{m}{T}\right). \tag{8.2.17}$$

The first term on the RHS of (8.2.17), $\frac{\sigma_a^2}{T} |G_T(f)|^2$, is a continuous dunction and its shape depends of $G_T(f)$.

The second term in (8.2.17) consists of discrete frequency components spaced $\frac{1}{T}$ apart in frequency.

Each component (or spectral line) has power that is proportional to $|G_T(f)|^2$ at $f = \frac{m}{T}$.

Note that the discrete frequency components can be eliminated by selecting the information symbol sequence $\{a_n\}$ to have zero mean.

The mean m_a in digital PAM, PSK, or QAM signals is easily forced to be zero by selecting the signal constellation points to be symmetrically positioned in the complex plane relative to the origin.

Under the condition that $m_a = 0$, the power-spectral density of the transmitted signal $V(t)$ becomes

$$S_V(f) = \frac{\sigma_a^2}{T} |G_T(f)|^2 . \tag{8.2.18}$$

Thus, the system designer can control the spectral characteristics of the transmitted digital PAM signal.

Ex. 8.2.1

Determine the power-spectral density of the transmitted signal $V(t)$ in (8.2.17), when $g_T(t)$ is the rectangular pulse shown in Figure 8.5(a).

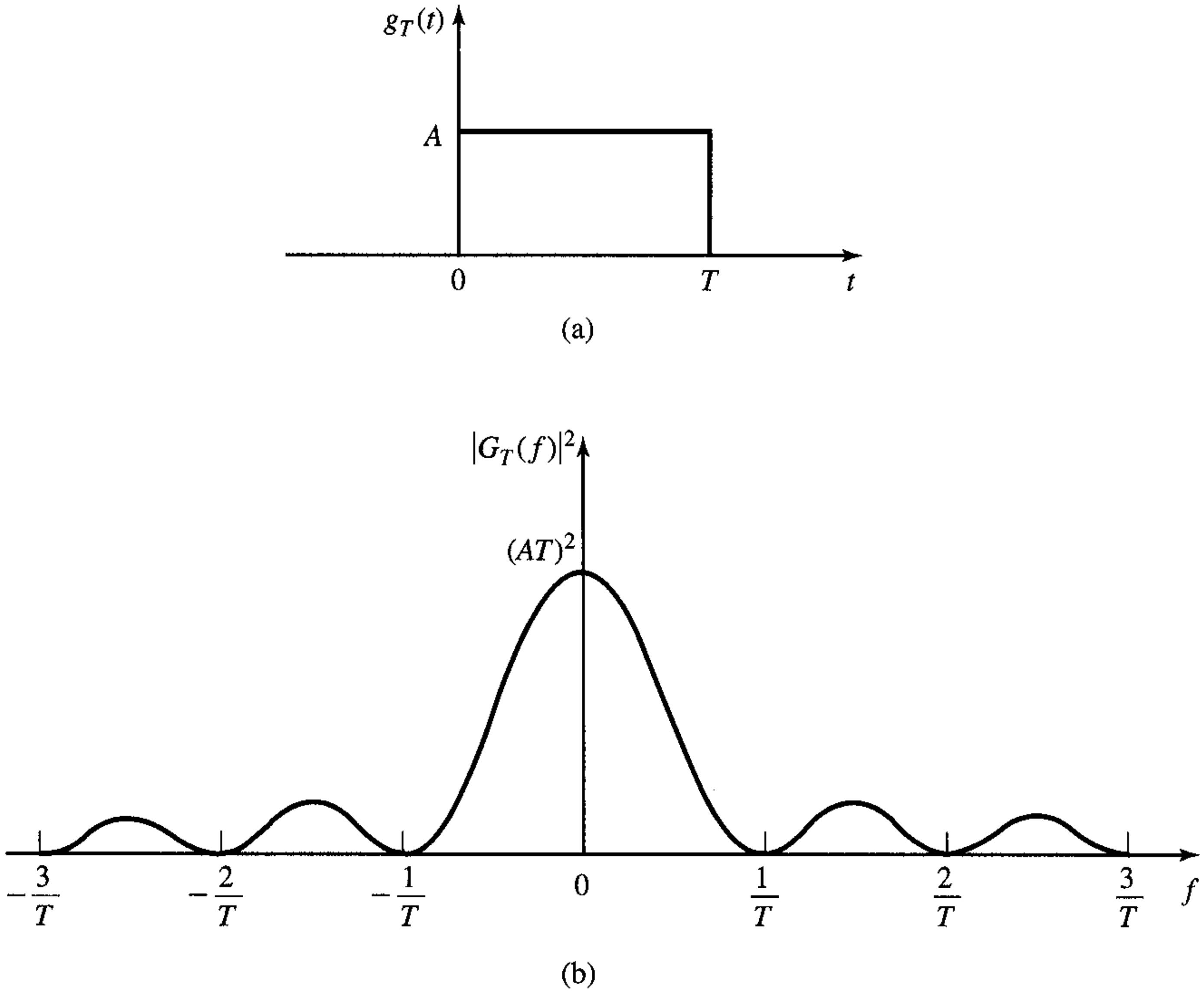


Figure 8.5 A rectangular pulse $g_T(t)$ and its energy density spectrum $|G_T(f)|^2$.

Solution

The Fourier transform of $g_T(t)$ is given by

$$G_T(f) = AT \frac{\sin \pi fT}{\pi fT} e^{-j\pi f T}.$$

Hence,

$$|G_T(f)|^2 = (AT)^2 \left(\frac{\sin \pi fT}{\pi fT} \right)^2$$

$$= (AT)^2 \operatorname{sinc}^2(fT)$$

which is shown in Figure 8.5(b).

Note that it contains nulls at multiples of $\frac{1}{T}$ in frequency and that it decays inversely as the square of the frequency variable.

As a consequence of the spectral nulls in $G_T(f)$, all but one of the discrete spectral components in (8.2.17) vanish.

Thus, upon substitution for $|G_T(f)|^2$ into (8.2.17), we obtain

$$S_V(f) = \sigma_a^2 A^2 T \left(\frac{\sin \pi fT}{\pi fT} \right)^2$$

$$= \sigma_a^2 A^2 T \operatorname{sinc}^2(fT) + A^2 m_a^2 \delta(f).$$

Ex. 8.2.2

Consider a binary sequence $\{b_n\}$, from which we form the symbols

$$a_n = b_n + b_{n-1}.$$

The $\{b_n\}$ are assumed to be uncorrelated binary valued (± 1) random variables, each having zero mean and unit variance.

Determine the power-spectral density of the transmitted signal.

Solution

The autocorrelation function of the sequence $\{a_n\}$ is given by

$$
\begin{aligned}
R_a(m) &= E[a_n a_{n+m}] \\
&= E[(b_n + b_{n-1})(b_{n+m} + b_{n+m-1})] \\
&= \begin{cases} 2, & m = 0, \\ 1, & m = \pm 1, \\ 0, & \text{otherwise.} \end{cases}
\end{aligned}
$$

Hence, the power-spectral density of the input sequence is given by

$$
\begin{aligned}
S_a(f) &= 2(1+\cos 2\pi fT) \\
&= 4\cos^2 \pi fT
\end{aligned}
$$

and, from (8.2.17), the corresponding power spectral density for the modulated signal is given by

$$S_V(f) = \frac{4}{T}|G_T(f)|^2 \cos^2 \pi fT \,.$$

Figure 8.6 shows the power-density spectrum $S_a(f)$ of the input sequence, and the corresponding $S_V(f)$ when $G_T(f)$ is the spectrum of the rectangular pulse.

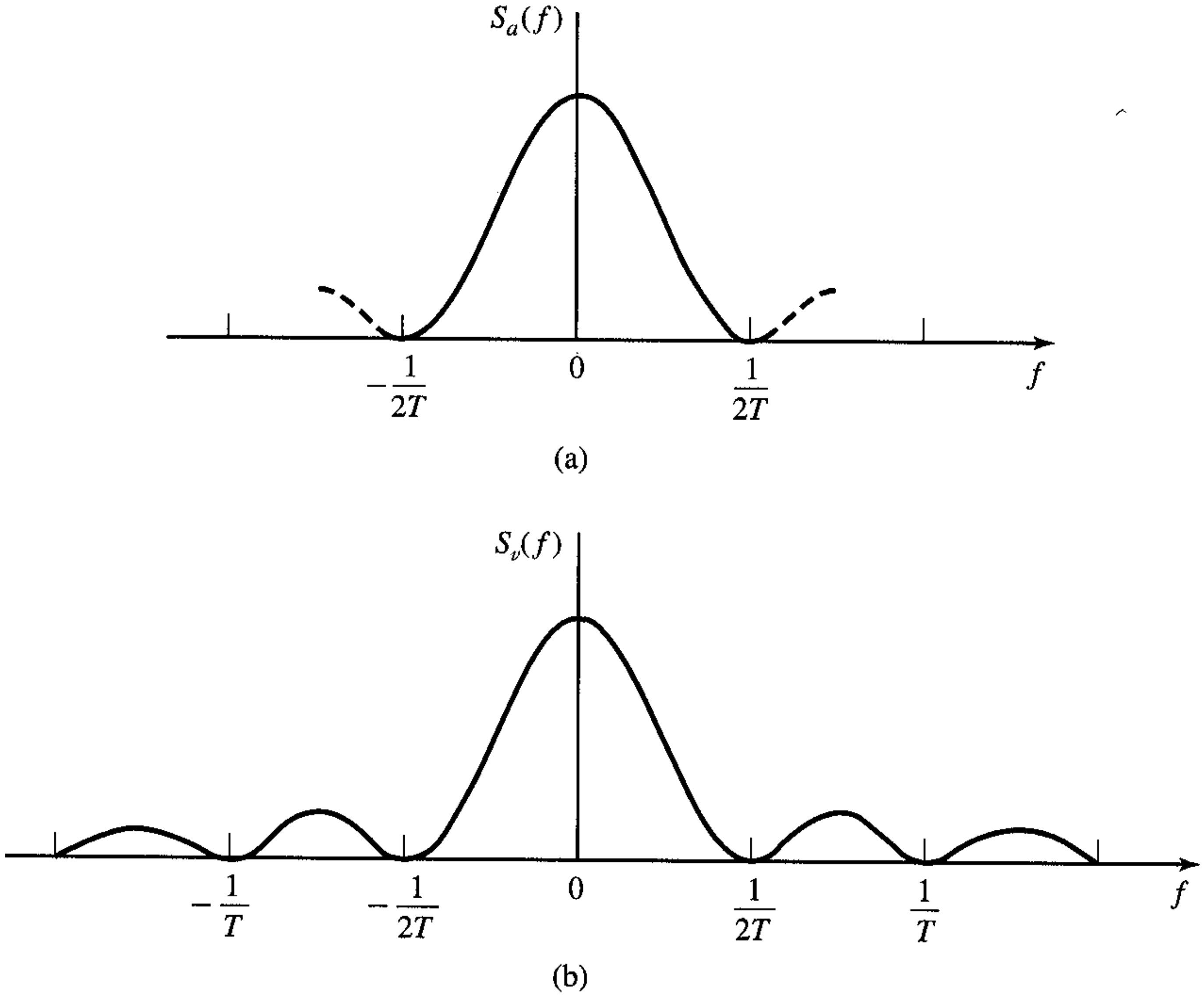


Figure 8.6 Power-density spectra for (a) information sequence and (b) PAM modulated signal.

As demonstrated in the example, the power spectral density of the transmitted signal can be shaped by having a correlated sequence $\{a_n\}$ as the input to the modulator.

8.2.2 Power Spectral Density of a Carrier-Modulated Signal

The autocorrelation function of the information sequence $\{a_n\}$ is given by

$$R_a(m) = E[a_n^* a_{n+m}]. \tag{8.2.21}$$

The power spectral density of the information sequence $\{a_n\}$ is given by

$$S_a(f) = \sum_{m=-\infty}^{\infty} R_a(m) e^{-j2\pi fmT}. \tag{8.2.20}$$

In Section 8.2.1, it was shown that the power spectral density of the equivalent baseband signal $v(t)$ in (8.2.1) for bandpass PAM, QAM, and PSK is given by

$$S_V(f) = \frac{1}{2} S_a(f) \,|\, G_T(f) \,|^2 . \tag{8.2.19}$$

To find out the relationship between the power spectral density of the baseband signal to the power spectral density of the bandpass signal, consider the bandpass PAM signal as an example.

The autocorrelation function of the bandpass signal

$$u(t) = v(t)\cos 2\pi f_c t$$

is given by

$$\begin{aligned} R_U(t+\tau, t) &= E[U(t+\tau)U(t)] \\ &= E[VV(t)]\cos 2\pi f_c(t+\tau)\cdot\cos 2\pi f_c t \\ &= R_V(t+\tau, t)\cos 2\pi f_c(t+\tau)\cdot\cos 2\pi f_c t \\ &= \frac{1}{2} R_V(t+\tau, t)[\cos 2\pi f_c\tau + \cos 2\pi f_c(2t+\tau)] . \end{aligned}$$

Then, the average of $R_U(t+\tau, t)$ over a single period T is given by

$$\overline{R}_U(\tau) = \frac{1}{2}\overline{R}_V(\tau)\cos 2\pi f_c\tau , \tag{8.2.22}$$

as the second term in the previous equation involving the double frequency averages to zero for each period of $\cos 4\pi f_c t$.

The power spectral density of the bandpass signal $u(t)$ is given by

$$S_U(f) = \int_{-\infty}^{\infty} \overline{R}_U(\tau) e^{-j2\pi f\tau} d\tau$$
$$= \frac{1}{4}[S_V(f - f_c) + S_V(f + f_c)]. \tag{8.2.23}$$

Although (8.2.23) was derived for PAM, it also applies to QAM and PSK.

The bandpass PAM, QAM, and PSK signals differ only in the autocorrelation function $R_a(m)$ of the sequence $\{a_n\}$ and, hence, in the power spectral density $S_a(f)$ of $\{a_n\}$.

8.3 Signal Design for Bandlimited Channels

Consider the problem of designing a bandlimited transmitting filter under thae condition that there is no channel distortion.

Since $H(f) = C(f)G_T(f)$, for distortion-free transmission the frequency response characteristic of the channel has a constant magnitude and a linear phase over the bandwidth of the transmitted signal, that is,

$$C(f) = \begin{cases} C_0 e^{-j2\pi f t_0}, & |f| \le W, \\ 0, & |f| > W, \end{cases} \tag{8.3.3}$$

where W is the available channel bandwidth,

t_0 represents an arbitrary finite delay, which we set to zero for convenience, and

C_0 is a constant gain factor which we set to unity for convenience.

Thus, the distortion-free channel, $H(f) = G_T(f)$ for $|f| \le W$ and zero for $|f| > W$.

Consequently, the matched filter has a frequency response $H^*(f) = G_T^*(f)$ and its output at the periodic sampling times $t = mT$ has the form

$$y(mT) = x(0)a_m + \sum_{\substack{n=-\infty \\ n \neq m}}^{\infty} a_n x(mT - nT) + v(mT) \tag{8.3.4}$$

or, simply

$$y_m = x_0 a_m + \sum_{\substack{n=-\infty \\ n \neq m}}^{\infty} a_n x_{m-n} + v_m \tag{8.3.5}$$

where $x(t) = g_T(t) * g_R(t)$ and

$v(t)$ is the output response of the matched filter to the input AWGN process $n(t)$.

The second term on the RHS of (8.3.5) represents the ISI.

The amount of ISI and noise in the received signal can be viewed on the vertical input of an oscilloscope.

The received signal is displayed on the vertical input with the horizontal sweep rate set at $\frac{1}{T}$,.

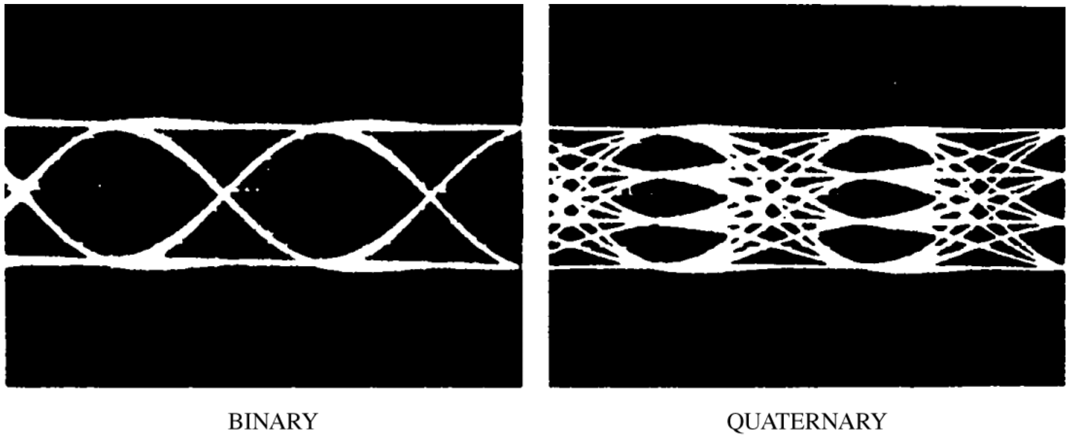
The resulting oscilloscope display is called an **eye pattern** because of its resemblance to the human eye.

Two examples of eye patterns for binary PAM and quaternary (or 4-ary, $M = 4$) PAM are shown in Figure 8.7(a).

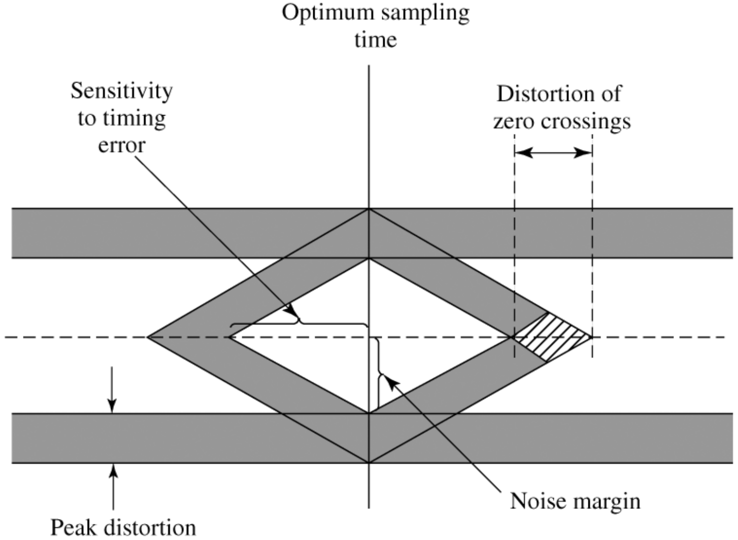
The effect of ISI is to cause the eye to close, thereby reducing the margin for additive noise to cause errors which is shown in Figure 8.7(b).

Note that ISI distorts the position of the zero crossings and causes a reduction in the eye opening.

As a consequence, the system is more sensitive to a synchronization error and has a smaller margin against additive noise.



(a)



(b)

Figure 8.7

Eye patterns. (a) Examples of eye patterns for binary and quaternary amplitude-shift keying (or PAM) and (b) Effect of ISI on eye opening.

8.3.1 Design of Bandlimited Signals for Zero ISI-The Nyquist Criterion

In a digital communication system over a bandlimited channel, the Fourier transform of the signal at the output of the receiving filter is given by

$$X(f) = G_T(f)C(f)G_R(f)$$

where $G_T(f)$ and $G_R(f)$ denote the transmitter and receiver filters frequency response, respectively, and $C(f)$ denotes the frequency response of the channel.

We have also seen that the output of the receiving filter, sampled at $t = mT$, is given by

$$y_m = x(0)a_m + \sum_{\substack{n=-\infty \\ n \neq m}}^{\infty} x(mT - nT)a_n + v(mT). \tag{8.3.6}$$

To remove the effect of ISI, it is necessary and sufficient that $x(mT - nT) = 0$ for $n \neq m$ and $x(0) \neq 0$, where we can assume $x(0) = 1$, without loss of generality.[†]

[†] The choice of $x(0)$ is equivalent to the choice of a constant gain factor in the receiving filter. This constant gain factor has no effect on the overall system performance since it scales both the signal and the noise.

This implies that the overall communication system has to be designed such that

$$x(nT) = \begin{cases} 1, & n = 0, \\ 0, & n \neq 0. \end{cases} \tag{8.3.7}$$

In this section, we derive the necessary and sufficient condition for $X(f)$ in order for $x(t)$ to satisfy the above relation which is known as the **Nyquist pulse-shaping criterion** **or** **Nyquist condition for zero ISI**.

Theorem 8.3.1 [Nyquist]

A necessary and sufficient condition for $x(t)$ to satisfy

$$x(nT) = \begin{cases} 1, & n = 0, \\ 0, & n \neq 0, \end{cases} \tag{8.3.8}$$

is that its Fourier transform $X(f)$ satisfy

$$\sum_{m=-\infty}^{\infty} X\left(f + \frac{m}{T} \right) = T\ . \tag{8.3.9}$$

Proof

In general, $x(t)$ is the inverse Fourier transform of $X(f)$.

Hence,

$$x(t) = \int_{-\infty}^{\infty} X(f) e^{j2\pi f t} df \ . \tag{8.3.10}$$

At the sampling instants $t = nT$, this relation becomes

$$x(nT) = \int_{-\infty}^{\infty} X(f) e^{j2\pi f nT} df \ . \tag{8.3.11}$$

Break up the integral in (8.3.11) into integrals covering the finite range of $\frac{1}{T}$.

Then, we obtain

$$\begin{aligned} x(nT) &= \sum_{m=-\infty}^{\infty} \int_{\frac{2m-1}{2T}}^{\frac{2m+1}{2T}} X(f) e^{j2\pi f nT} df \\ &= \sum_{m=-\infty}^{\infty} \int_{-\frac{1}{2T}}^{\frac{1}{2T}} X\left(f + \frac{m}{T} \right) e^{j2\pi f nT} dt \\ &= \int_{-\frac{1}{2T}}^{\frac{1}{2T}} \left[\sum_{m=-\infty}^{\infty} X\left(f + \frac{m}{T} \right) \right] e^{j2\pi f nT} dt \\ &= \int_{-\frac{1}{2T}}^{\frac{1}{2T}} Z(f) e^{j2\pi f nT} dt \end{aligned} \tag{8.3.12}$$

where

$$Z(f) = \sum_{m=-\infty}^{\infty} X\left(f + \frac{m}{T} \right). \tag{8.3.13}$$

As $Z(f)$ is a periodic function with period $\frac{1}{T}$, it can be expanded in terms of its Fourier series coefficients $\{z_n\}$ as

$$Z(f) = \sum_{m=-\infty}^{\infty} z_n e^{j2\pi nfT} \tag{8.3.14}$$

where

$$z_n = T\int_{-\frac{1}{2T}}^{\frac{1}{2T}} Z(f) e^{-j2\pi nfT} df \ . \tag{8.3.15}$$

Comparing (8.3.15) and (8.3.12) we obtain

$$z_n = T\ x(-nT)\,. \tag{8.3.16}$$

Therefore, the necessary and sufficient conditions for (8.3.8) to be satisfied is that

$$z_n = \begin{cases} T, & n = 0, \\ 0, & n \neq 0, \end{cases} \tag{8.3.17}$$

which, when substituted into (8.3.14), yields

$$Z(f) = T \tag{8.3.18}$$

or, equivalently,

$$\sum_{m=-\infty}^{\infty} X\left(f + \frac{m}{T} \right) = T \tag{8.3.19}$$

which concludes the proof of the theorem.

Now, suppose that the channel has a bandwidth of W.

Then, $C(f) \equiv 0$ for $|f| > W$ and consequently, $X(f) = 0$ for $|f| > W$. ($X(f) = G_T(f)C(f)G_R(f)$)

We distinguish three cases:

1. $T < \frac{1}{2W}$, or equivalently, $\frac{1}{T} > 2W$.

Since $Z(f) = \sum_{n=-\infty}^{+\infty} X\left(f + \frac{n}{T}\right)$ consists of nonoverlapping replicas of $X(f)$, separated by $\frac{1}{T}$ as shown in Figure 8.8, there is no choice for $X(f)$ to achieve $Z(f) \equiv T$, and there is no way to design a system with no ISI.

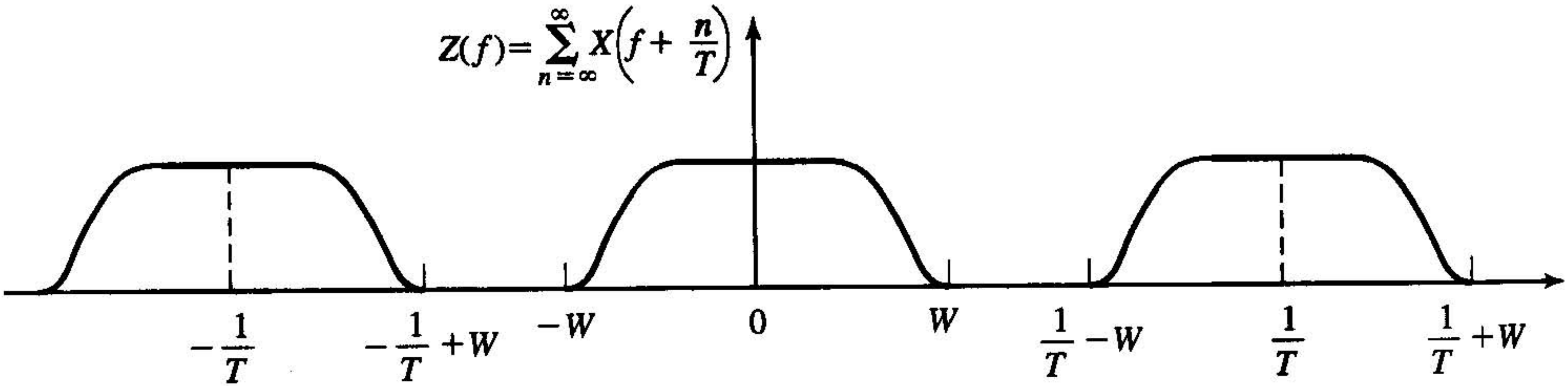


Figure 8.8 Plot of $Z(f)$ for the case $T < \frac{1}{2W}$.

2. $T = \dfrac{1}{2W}$, or equivalently, $\dfrac{1}{T} = 2W$ (the Nyquist rate).

The replicas of $X(f)$, separated by $\dfrac{1}{T}$, are about to overlap as shown in Figure 8.9.

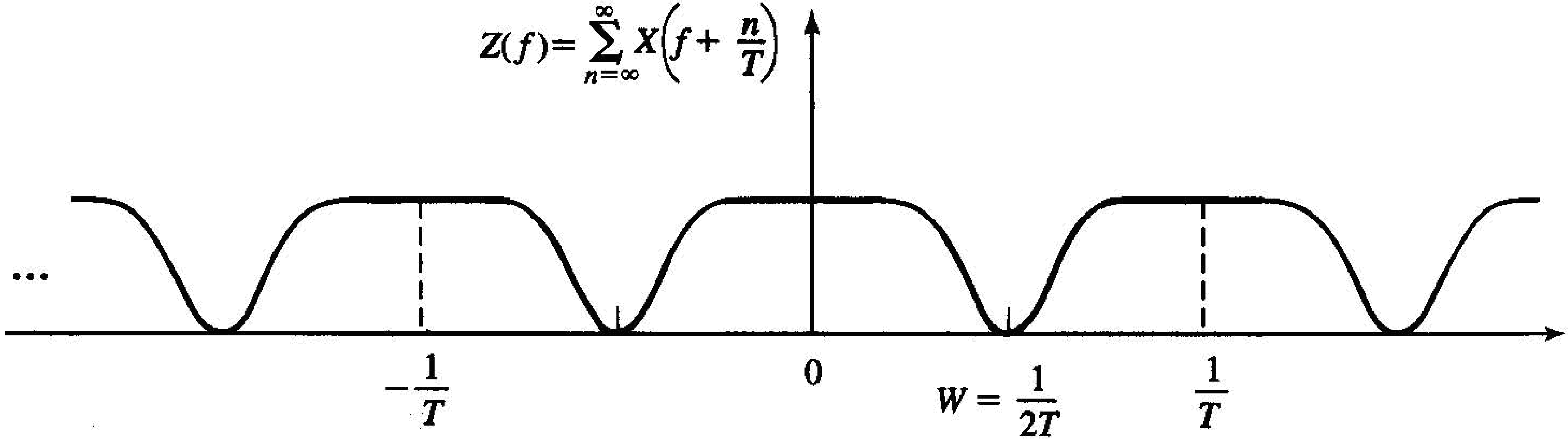


Figure 8.9 Plot of $Z(f)$ for the case $T = \frac{1}{2W}$.

It is clear that there exists only one $X(f)$ that results in $Z(f) = T$, that is,

$$X(f) = \begin{cases} T, & |f| < W, \\ 0, & \text{otherwise}, \end{cases} \tag{8.3.20}$$

or, $X(f) = T\,\Pi\left(\frac{f}{2W}\right)$, which results in

$$x(t) = \text{sinc}\left(\frac{t}{T}\right). \tag{8.3.21}$$

This impliess that the smallest value of T for which transmission with zero ISI possible is $T = \frac{1}{2W}$ and $x(t)$ has to be a sinc function.

The difficulty with this choice of $x(t)$ is that it is non-causal and therefore non-realizable.

To make it realizable, usually a delayed version of it, that is, $\text{sinc}\left(\frac{t-t_0}{T}\right)$ is used and t_0 is chosen large enough such that for $t < 0$, we have $\text{sinc}\left(\frac{t-t_0}{T}\right) \approx 0$.

With this choice of $x(t)$, the sampling time must also be shifted to $mT + t_0$.

A second difficulty with this pulse shape is that its rate of convergence to zero is slow.

The tails of $x(t)$ decay as $\frac{1}{t}$, consequently, a small mistiming error in sampling the output of the matched filter at the demodulator results in an infinite series of ISI components.

Such a series is not absolutely summable because of the $\frac{1}{t}$ rate of decay of the pulse and, hence, the sum of the resulting ISI does not converge.

3. For $T > \frac{1}{2W}$, $Z(f)$ consists of overlapping replications of $X(f)$ separated by $\frac{1}{T}$, as shown in Figure 8.10.

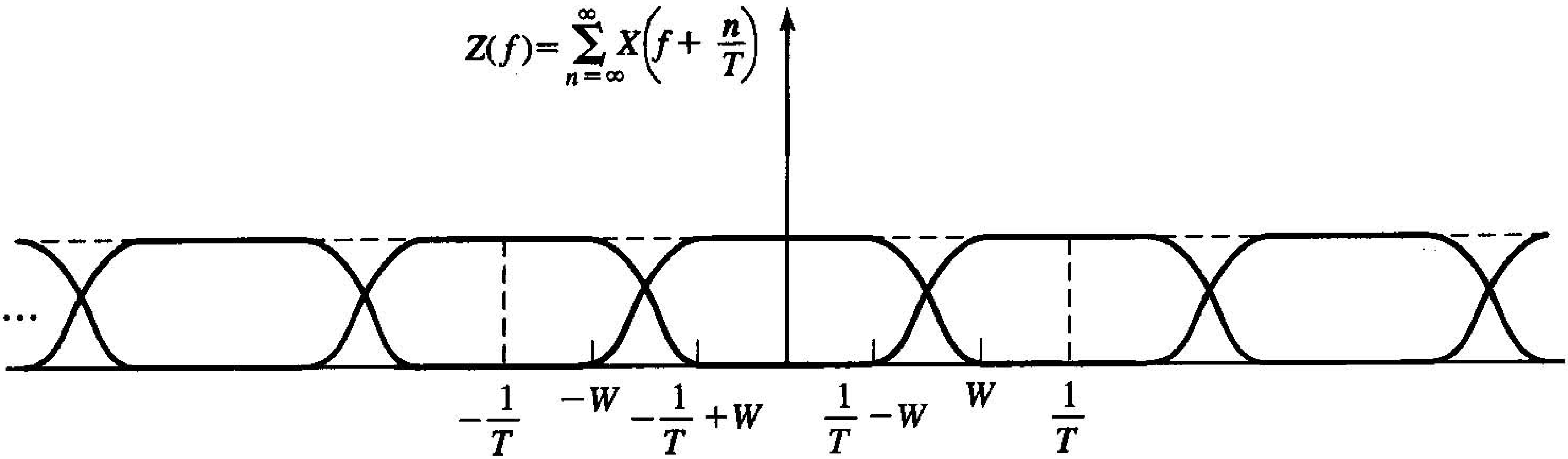


Figure 8.10 Plot of $Z(f)$ for the case $T > 1/2W$.

There exist numerous choices for $X(f)$, such that $Z(f) \equiv T$.

For the $T > \frac{1}{2W}$ case, a particular pulse spectrum that has been widely used is the raised cosine spectrum of which frequency characteristic is given as (see Problem 8.11)

$$X_{rc}(f) = \begin{cases} T, & 0 \le |f| \le \frac{1-\alpha}{2T}, \\ \frac{T}{2}\left[1+\cos\frac{\pi T}{\alpha}\left(|f| - \frac{1-\alpha}{2T}\right)\right], & \frac{1-\alpha}{2T} \le |f| \le \frac{1+\alpha}{2T}, \\ 0, & |f| > \frac{1+\alpha}{2T}, \end{cases} \tag{8.3.22}$$

where α is called the **rolloff factor**, $0 \le \alpha \le 1$.

The bandwidth occupied by the signal beyond the Nyquist frequency $\frac{1}{2T}$ is called the **excess bandwidth** and is usually expressed as a percentage of the Nyquist frequency.

For example, when $\alpha = \frac{1}{2}$, the excess bandwidth is 50%, and when $\alpha = 1$ the excess bandwidth is 100%.

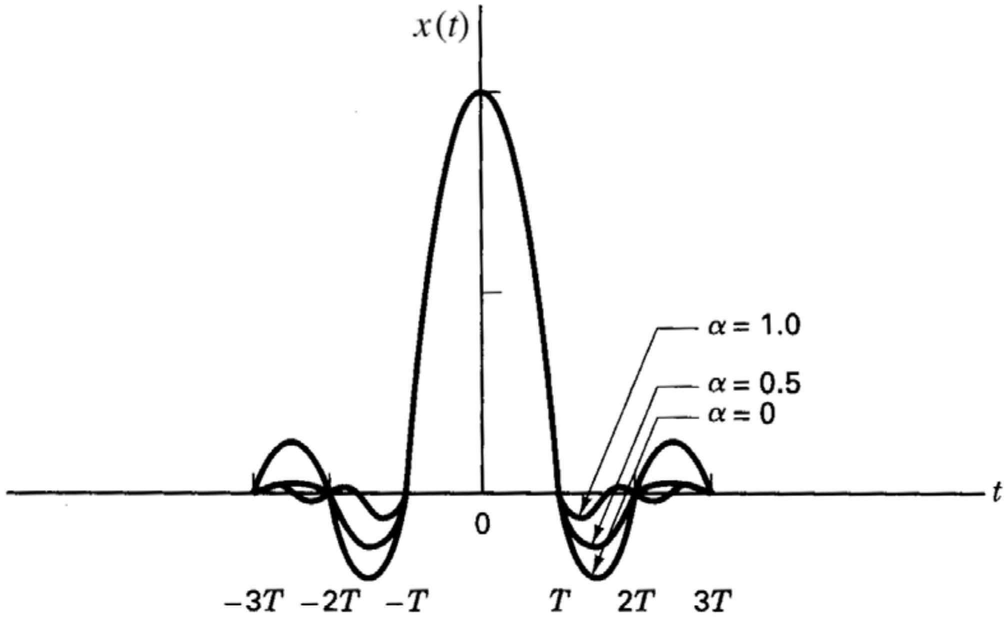
The pulse having the raised cosine spectrum is given by

$$x(t) = \frac{\sin\frac{\pi t}{T}}{\frac{\pi t}{T}} \cdot \frac{\cos\frac{\pi\alpha t}{T}}{1 - \frac{4\alpha^2 t^2}{T^2}}$$

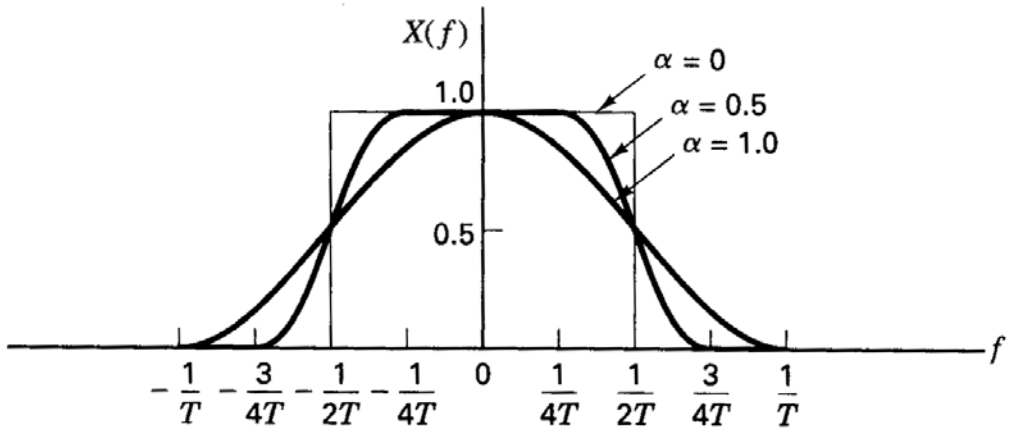
$$= \mathrm{sinc}(\frac{t}{T}) \cdot \frac{\cos \dfrac{\pi\alpha\, t}{T}}{1 - \dfrac{4\alpha^2\, t^2}{T^2}}. \tag{8.3.23}$$

Note that $x(t)$ is normalized so that $x(0) = 1$.

Figure 8.11 shows the raised cosine spectral characteristics and the corresponding pulse for $\alpha = 0, \frac{1}{2}, 1$.



(a)



(b)

Figure 8.11 Pulses having a raised cosine spectrum.

Note that for $\alpha = 0$, the pulse reduces to $x(t) = \text{sinc}\left(\frac{t}{T}\right)$, and the symbol rate $\frac{1}{T} = 2W$.

When $\alpha = 1$, the symbol rate is $\frac{1}{T} = W$.

In general, the tails of $x(t)$ decay as $\frac{1}{t^3}$ for $\alpha > 0$.

Consequently, a mistiming error in sampling leads to a series of intersymbol interference components that converges to a finite value.

Thanks to smooth characteristics of the raised cosine spectrum, it is possible to design practical filters for the transmitter and receiver that approximate the overall desired frequency responseing filter

In this case, if the receiver filter is matched to the transmitting filter, we have

$$X_{rc}(f) = G_T(f)G_R(f)$$

$$= |G_T(f)|^2.$$

Ideally,

$$G_T(f) = \sqrt{|X_{rc}(f)|}\, e^{-j2\pi f t_0} \tag{8.3.25}$$

and $G_R(f) = G_T^*(f)$,

where t_0 is some nominal delay that is required to assure physical realizability of the filter.

Thus, the overall raised cosine spectral characteristic is split evenly between the transmitting filter and the receiving filter.

Also note that an additional delay is necessary to ensure the physical realizability of the receiving filter.

8.3.2 Design of Bandlimited Signals with Controlled ISI-Partial Response Signals

To achieve zero ISI, it is necessary to reduce the symbol rate $\frac{1}{T}$ below the Nyquist rate of $2W$ symbols/sec in order to realize practical transmitting and receiving filters.

By relaxing the condition of zero ISI that $x(nT) = 0$ for $n \neq 0$, we can achieve a symbol transmission rate of $2W$ symbols/sec.

Suppose that we design the bandlimited signal to have controlled ISI at one time instant which implies that we allow one additional nonzero value in the samples $\{x(nT)\}$.

The ISI that we introduce is deterministic or "controlled" and, hence, it can be taken into account at the receiver.

One special case that leads to (approximately) physically realizable transmitting and receiving filters is specified by the samples‡

$$x(nT) = \begin{cases} 1, & n = 0, \\ 1, & n = 1, \\ 0, & \text{otherwise.} \end{cases} \tag{8.3.26}$$

By using (8.3.16), we obtain

$$z_n = \begin{cases} T, & n = -1, \\ T, & n = 0, \\ 0, & \text{otherwise} \end{cases} \tag{8.3.27}$$

From (8.3.14) and (8.3.27) we have

$$Z(f) = T + Te^{-j2\pi fT} \tag{8.3.28}$$

It is impossible to satisfy (8.3.28) for $T < \dfrac{1}{2W}$.

‡ It is convenient to deal with samples of $x(t)$ that are normalized to unity for $n = 0,1$.

However, for $T = \dfrac{1}{2W}$, we obtain

$$X(f) = \begin{cases} \dfrac{1}{2W}\left[1 + e^{-j\frac{\pi f}{W}}\right], & |f| < W, \\ 0, & \text{otherwise}, \end{cases}$$

$$= \begin{cases} \dfrac{1}{W} e^{-j\frac{\pi f}{2W}} \cos\dfrac{\pi f}{2W}, & |f| < W, \\ 0, & \text{otherwise}. \end{cases} \tag{8.3.29}$$

Therefore, $x(t)$ is given by

$$x(t) = \text{sinc}(2Wt) + \text{sinc}(2Wt - 1) \tag{8.3.30}$$

which is called a **duobinary signal pulse** shown in Figure 8.12.

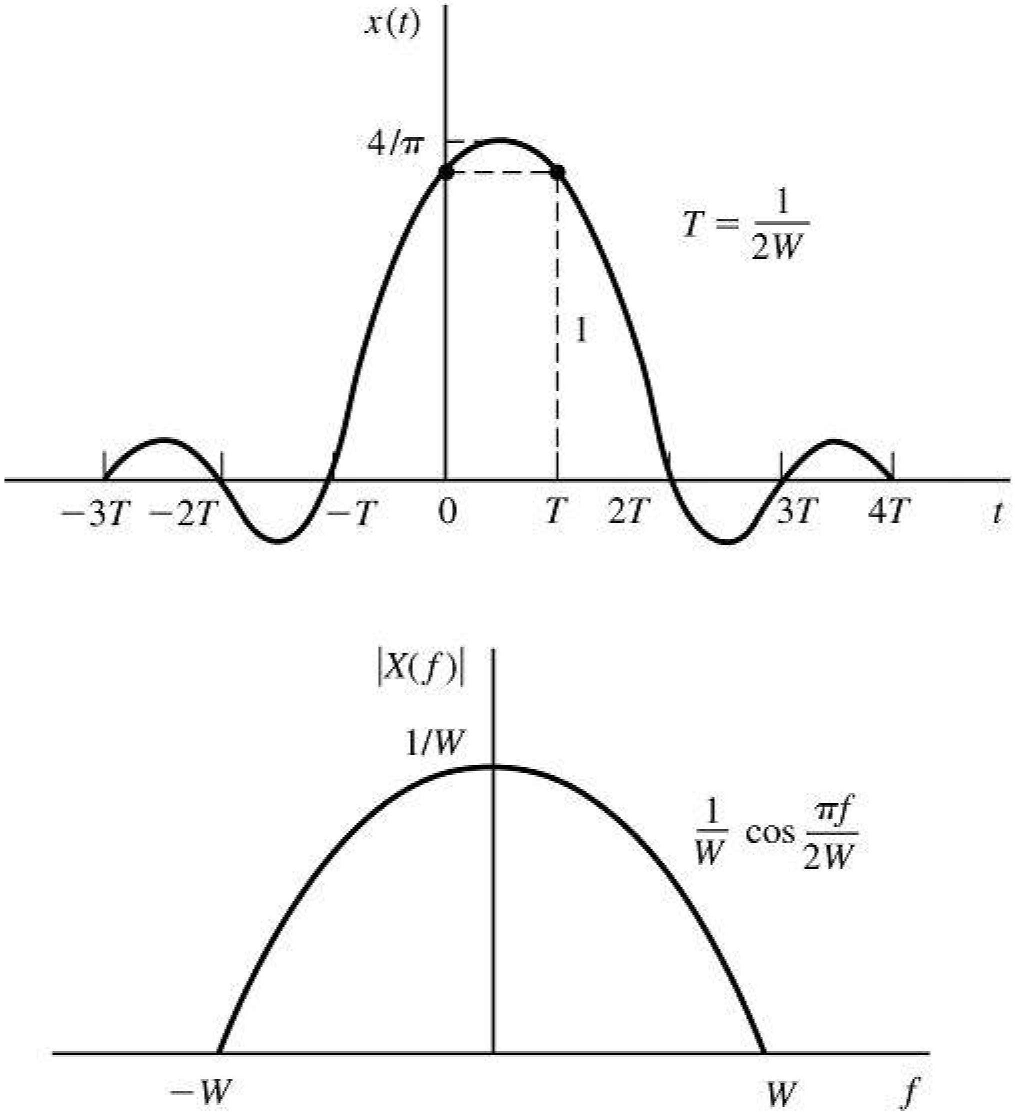


Figure 8.12

Time domain and frequency-domain characteristics of a duobionary signal.

Note that the spectraum decays to zero smoothly, which means that physically realizable filters can be designed that approximate this spectrum very closely.

Thus, a symbol rate of $2W$ is achieved.

Another special case that leads to (approximately) physically realizable transmitting and receiving filters is specified by the samples

$$x\left(\frac{n}{2W}\right) = x(nT)$$

$$= \begin{cases} 1, & n = -1, \\ -1, & n = 1, \\ 0, & \text{otherwise.} \end{cases} \tag{8.3.31}$$

The corresponding pulse $x(t)$ is given by

$$x(t) = \mathrm{sinc}(\frac{t+T}{T}) - \mathrm{sinc}(\frac{t-T}{T}) \tag{8.3.32}$$

and its spectrum is given by

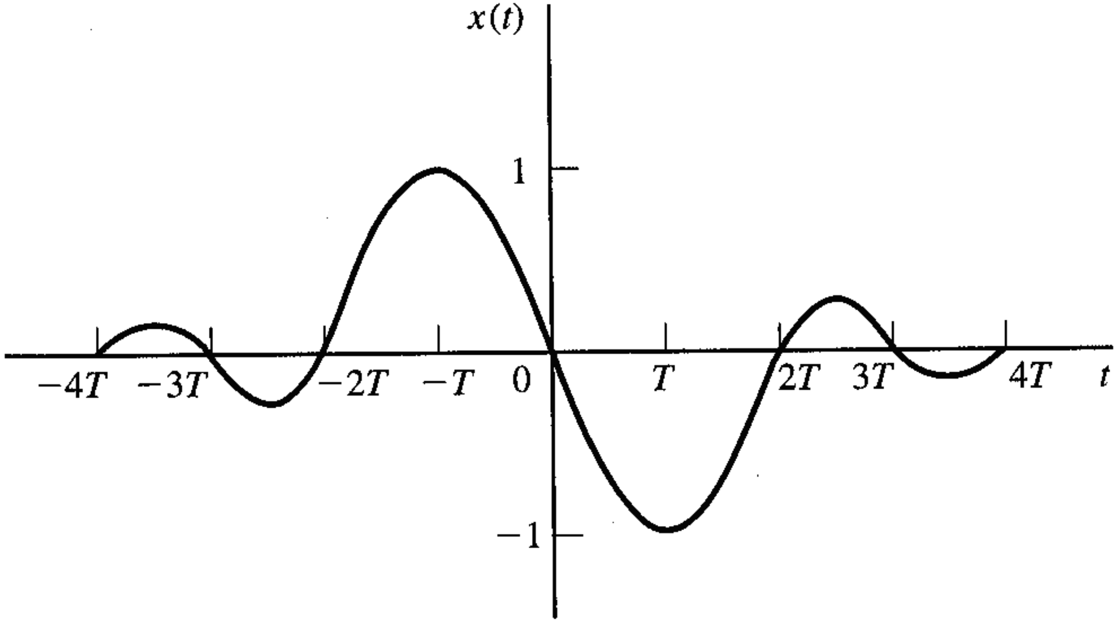
$$X(f) = \begin{cases} \dfrac{1}{2W} e^{j\frac{\pi f}{W}} - e^{j\frac{\pi f}{W}}, & |f| \le W, \\ 0, & |f| > W, \end{cases}$$

$$= \begin{cases} \dfrac{j}{W} \sin \dfrac{\pi f}{W}, & |f| \le W, \\ 0, & |f| > W, \end{cases} \tag{8.3.33}$$

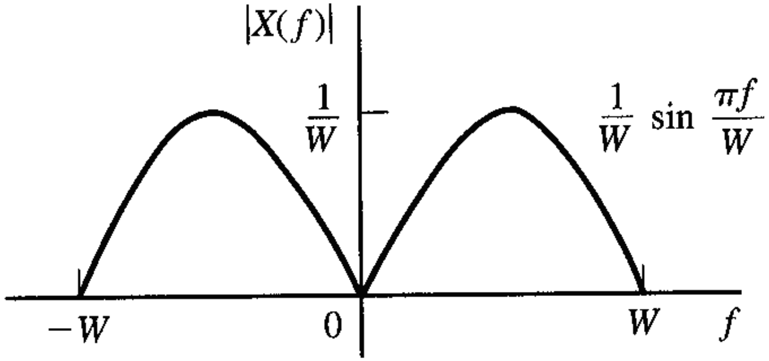
which is shown in Figure 8.13.

It is called a **modified duobinary signal pulse**.

Note that the spectrum of this signal has a zero at $f = 0$, making it suitable for transmission over a channel that does not pass D.C.



(a)



(b)

Figure 8.13 Time domain and frequency domain characteristics of a modified duobinary signal.

One can obtain other interesting and physically realizable filter characteristics by selecting different values for the samples $\left\{ x(\frac{n}{2W}) \right\}$ and more than two nonzero samples.

However, as we select more nonzero samples, solving the controlled ISI becomes more cumbersome and impractical.

In general, the class of bandlimited signals pulses that have the form

$$x(t) = \sum_{n=-\infty}^{\infty} x(\frac{n}{2W}) \cdot \frac{\sin 2\pi W(t - \frac{n}{2W})}{2\pi W(t - \frac{n}{2W})} \tag{8.3.34}$$

and their corresponding spectra

$$X(f) = \begin{cases} \dfrac{1}{2W} \displaystyle\sum_{n=-\infty}^{\infty} x(\frac{n}{2W}) e^{-jn\pi f}, & |f| \le W, \\ 0, & |f| > W, \end{cases} \tag{8.3.35}$$

are called **partial response signals** when controlled ISI is ISI is purposely introduced by selecting two or more nonzero samples from the set $\left\{ x(\frac{n}{2W}) \right\}$.

The resulting signal pulses allow us to transmit information symbols at the Nyquist rate of $2W$ symbols per second.

8.4 Probability of Error in Detection of PAM

We evaluate the performance of the receiver for demodulating and detecting an M-ary PAM signal in the presence of additive white Gaussian noise at its input for two cases:

1) the case that the transmitting and receiving filters $G_T(f)$ and $G_R(f)$ are designed for zero ISI and

2) the case that $G_T(f)$ and $G_R(f)$ are designed such that $x(t) = g_T(t) * g_R(t)$ is either a duobinary signal or a modified duobinary signal.

The results can be generalized to two-dimensional modulations, such as PSK and QAM, and multidimensional signals.

8.4.1 Probability of Error for Detection of PAM with Zero ISI

In the absence of ISI, the received signal sample at the output of the receiving matched filter is given by

$$y_m = x_0 a_m + v_m \tag{8.4.1}$$

where

$$x_0 = \int_{-W}^{W} |G_T(f)|^2 \, df$$
$$= \varepsilon_g \tag{8.4.2}$$

and v_m is the additive Gaussian noise which has zero mean and variance

$$\sigma_v^2 = \frac{\varepsilon_g N_0}{2}. \tag{8.4.3}$$

Suppose that a_m takes one of M possible equally spaced amplitude values with equal probability.

Then, the probability of error for digital PAM in a bandlimited additive white Gaussian noise channel in the absence of ISI is identical to that for M-ary PAM as given in Section 7.6.2.

The probability of error for digital PAM is given by

$$P_M = \frac{2(M-1)}{M} Q\left(\sqrt{\frac{2\varepsilon_g}{N_0}}\right). \tag{8.4.4}$$

Since $\varepsilon_g = \dfrac{3\varepsilon_{av}}{M^2-1}$, where $\varepsilon_{av} = k\varepsilon_{bav}$ is the average energy per symbol and ε_{bav} is the average energy/bit, (8.4.4) becomes

$$P_M = \frac{2(M-1)}{M} Q\left(\sqrt{\frac{6(\log_2 M)\varepsilon_{bav}}{(M^2-1)N_0}}\right) \tag{8.4.5}$$

which is exactly the form for the probability of error of M -ary PAM derived in Section 7.6.2.

While there was no bandwidth constraint on the PAM signals in Section 7.6.2, here the transmitted signal pulses are designed to be bandlimited and to have zero ISI.

Nevertheless, the receivers (demodulators and detectors) in both cases are optimum (matched filters) for the corresponding transmitted signals.

Consequently, no loss in error-rate performance results from the bandwidth constraint when the signal pulse is designed for zero ISI and the channel does not distort the transmitted signal.

8.4.2 Symbol-by-Symbol Detection of Data with Controlled ISI

Consider a symbol-by-symbol method for detecting the information symbols at the demodulator when the received signal contains controlled ISI.

In particular, we consider the detection of the duobinary and the modified duobinary partial response signals.

In both cases, assume that the desired spectral characteristic $X(f)$ for the partial response signal is split evenly between the transmitting and receiving filters, that is, $|G_T(f)| = |G_R(f)| = v|X(f)|^{\frac{1}{2}}$.

For the duobinary signal pulse, $x(nT) = \begin{cases} 1, & n = 0, \\ 1, & n = 1, \\ 0, & \text{otherwise.} \end{cases}$

Hence, the samples at the output of the receiving filter are given by

$$\begin{aligned} y_m &= b_m + v_m \\ &= a_m + a_{m-1} + v_m \end{aligned} \tag{8.4.6}$$

where $\{a_m\}$ is the transmitted sequence of amplitudes and

$\{v_m\}$ is a sequence of additive Gaussian noise samples.

Ignore the noise for the moment and consider the binary case where $a_m = \pm 1$ with equal probability.

Then, b_m takes on one of three possible values, namely, $b_m = -2, 0, 2$ with corresponding probabilities $\frac{1}{4}, \frac{1}{2}, \frac{1}{4}$.

If the detected symbol from the $(m-1)$st signaling interval is a_{m-1}, its effect on the received signal in the mth signaling interval b_m can be eliminated by subtraction, thus allowing a_m to be detected.

This process can be repeated sequentially for every received symbol.

The major problem with this procedure is that errors arising from the additive noise tend to propagate.

That is, if a_{m-1} is in error, its effect on b_m is not eliminated but, in fact, it is reinforced by the incorrect subtraction so that the detection of a_m is also likely to be in error.

Error propagation can be avoided by **precoding** the data on the binary data sequence prior to modulation at the transmitter instead of eliminating the controlled ISI by subtraction at the receiver.

From the data sequence $\{d_n\}$ of 1's and 0's that is to be transmitted, a new sequence $\{p_n\}$, called the **precoded sequence,** is generated.

For the duobinary signal, the precoded sequence is given by

$$p_m = d_m \ominus p_{m-1}, \qquad m = 1, 2, \cdots, \tag{8.4.7}$$

where $\ominus$ denotes modulo-2 subtraction.†

Then, set $a_m = -1$ if $p_m = 0$ and $a_m = 1$ if $p_m = 1$, that is, $a_m = 2p_m - 1$.

The noise-free samples at the output of the receiving filter are given by

† Although this is identical to modulo-2 addition, it is convenient to view the precoding operation for duobinary in terms of modulo-2 subtraction.

$$
\begin{aligned}
b_m &= a_m + a_{m-1} \\
&= (2p_m - 1) + (2p_{m-1} - 1) \\
&= 2(p_m + p_{m-1} - 1). \qquad (8.4.8)
\end{aligned}
$$

Consequently, it becomes

$$p_m + p_{m-1} = \frac{b_m}{2} + 1. \qquad (8.4.9)$$

Since $d_m = p_m \oplus p_{m-1}$, it follows that

$$d_m = \frac{b_m}{2} + 1 \pmod{2}. \qquad (8.4.10)$$

Consequently, if $b_m = \pm 2$, $d_m = 0$ and if $b_m = 0$, $d_m = 1$.

An example that illustrates the precoding and decoding operations is given in Table 8.1.

Table 8.1 Binary Signaling with Duobinary Pulses.

Data sequence d_n		1	1	1	0	1	0	0	1	0	0	0	1	1	0	1
Precoded sequence p_n	0	1	0	1	1	0	0	0	1	1	1	1	0	1	1	0
Transmitted sequence a_n	−1	1	−1	1	1	−1	−1	−1	1	1	1	1	−1	1	1	−1
Received sequence b_n		0	0	0	2	0	−2	−2	0	2	2	2	0	0	2	0
Decoded sequence d_n		1	1	1	0	1	0	0	1	0	0	0	1	1	0	1

In the presence of additive noise the sampled outputs from the receiving filter are given by (8.4.6).

In this case $y_m = b_m + v_m$ is compared with the two thresholds set at $+1$ and -1.

The data sequence $\{d_n\}$ is obtained according to the detection rule

$$d_m = \begin{cases} 1, & -1 < y_m < 1, \\ 0, & |y_m| \geq 1. \end{cases} \tag{8.4.11}$$

The extension from binary PAM to multilevel PAM signaling using the duobinary pulse is straightforward.

In this case the M -level amplitude sequence $\{a_m\}$ results in a (noise-free) sequence

$$b_m = a_m + a_{m-1}, \qquad m = 1, 2, \cdots, \tag{8.4.12}$$

which has $2M-1$ possible equally spaced levels.

The amplitude levels are determined from the relation

$$a_m = 2p_m - (M-1) \tag{8.4.13}$$

where $\{p_m\}$ is the precoded sequence that is obtained from an M -level data sequence $\{d_m\}$ according to the relation (see (8.4.7))

$$p_m = d_m \ominus p_{m-1} (\text{mod } M) \tag{8.4.14}$$

where $\ominus$ denotes modulo-M subtraction and the possible values of the sequence $\{d_m\}$ are $0, 1, 2, \cdots, M-1$.

In the absence of noise, the samples at the output of the receiving filter are expressed as

$$
\begin{aligned}
b_m &= a_m + a_{m-1} \\
&= 2[p_m + p_{m-1} - (M-1)].
\end{aligned}
\tag{8.4.15}
$$

Hence,

$$
p_m + p_{m-1} = \frac{b_m}{2} + (M-1). \tag{8.4.16}
$$

Since $d_m = p_m + p_{m-1} \pmod{M}$, it follows that

$$
d_m = \frac{b_m}{2} + (M-1) \pmod{M}. \tag{8.4.17}
$$

An example illustrating multilevel precoding and decoding is given in Table 8.2.

Table 8.2 Four-Level Transmission with Duobinary Pulses.

Data sequence d_n		0	0	1	3	1	2	0	3	3	2	0	1	0
Precoded sequence p_n	0	0	0	1	2	3	3	1	2	1	1	3	2	2
Transmitted sequence a_n	−3	−3	−3	−1	1	3	3	−1	1	−1	−1	3	1	1
Received sequence b_n		−6	−6	−4	0	4	6	2	0	0	−2	2	4	2
Decoded sequence d_n		0	0	1	3	1	2	0	3	3	2	0	1	0

In the presence of noise, the received signal-plus-noise is quantized to the nearest of the possible signal levels and the rule given above is used on the quantized values to recover the data sequence.

In the case of the modified duobinary pulse, the controlled ISI is given by

$$x\left(\frac{n}{2W}\right)=\begin{cases}1, & n=-1,\\ -1, & n=1,\\ 0, & \text{otherwise.}\end{cases}$$

Consequently, the noise-free sampled output from the receiving filter is given by

$$b_m = a_m - a_{m-2} \tag{8.4.18}$$

where the M-level sequence $\{a_n\}$ is obtained by mapping a precoded sequence according to the relation (8.2.43) and

$$p_m = d_m \oplus p_{m-2} \ (\mathrm{mod}\ M). \tag{8.4.19}$$

From these relations, it is shown that the detection rule for recovering the data sequence $\{d_m\}$ from $\{b_m\}$ in the absence of noise is given by

$$d_m = \frac{b_m}{2} \ (\mathrm{mod}\ M). \tag{8.4.20}$$

Note that the precoding of the data at the transmitter makes it possible to detect the received data on a symbol-by-symbol basis without having to look back at previously detected symbols.

Thus, error propagation is avoided.

The symbol-by-symbol detection rule described above is not the optimum detection scheme for partial response signals.

Nevertheless, symbol-by-symbol detection is relatively simple to implement and is used in many practical applications involving duobinary and modified duobinary pulse signals.

8.4.3 Probability of Error for Detection of Partial Response Signals

Assume an ideal bandlimited channel with additive white Gaussian noise,

The model for the communications system is shown in Figure 8.14.

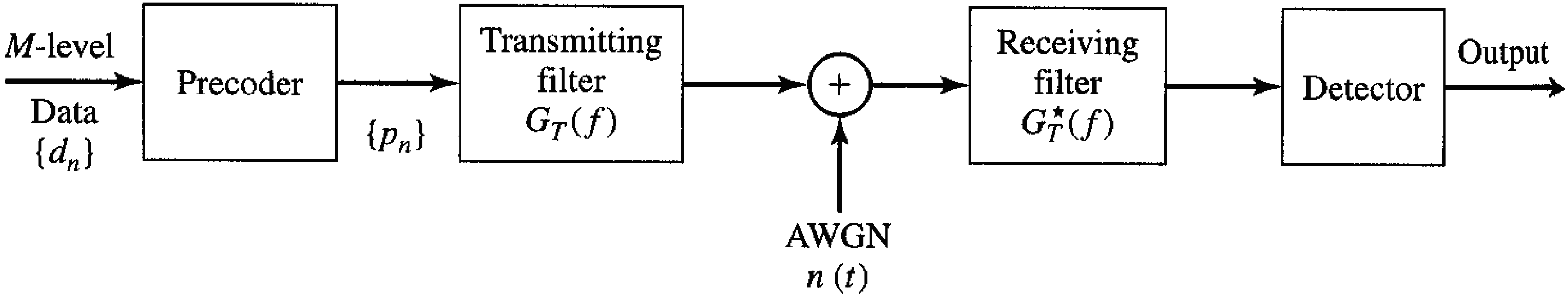


Figure 8.14 Block diagram of modulator and demodulator for partial response signals.

Symbol-by-Symbol Detector

At the transmitter, the M -level data sequence $\{d_n\}$ is precoded as described previously.

The precoder output is mapped into one of M possible amplitude levels.

Then the transmitting filter with frequency response $G_T(f)$ has an output

$$v(t) = \sum_{n=-\infty}^{\infty} a_n g_T(t - nT). \tag{8.4.21}$$

The partial-response function $X(f)$ is divided equally between the transmitting and receiving filters.

Hence, the receiving filter is matched to the transmitted pulse, and the cascade of the two filters results in the frequency characteristic

$$|G_T(f)G_R(f)| = |X(f)|. \tag{8.4.22}$$

The matched filter output is sampled at $t = nT = \frac{2}{2W}$ and the samples are fed to the decoder.

For the duobinary signal, the output of the matched filter at the sampling instant is given by

$$
\begin{aligned}
y_m &= a_m + a_{m-1} + v_m \\
&= b_m + v_m
\end{aligned}
\tag{8.4.23}
$$

where v_m is the additive noise component.

Similarly, the output of the matched filter for the modified duobinary signal is given by

$$
\begin{aligned}
y_m &= a_m - a_{m-2} + v_m \\
&= b_m + v_m .
\end{aligned}
\tag{8.4.24}
$$

For binary transmission, let $a_m = \pm d$, where $2d$ is the distance between signal levels.

Then, the corresponding values of b_m are $(2d, 0, -2d)$.

For M-ary PAM signal transmission, where $a_m = \pm d, \pm 3d, \cdots, \pm (M-1)d$, the received signal levels are given by $b_m = 0, \pm 2d \pm 4d, \cdots, \pm 2(M-1)d$.

Hence, the number of received levels is $2M-1$.

Assume that the input transmitted symbol $\{a_m\}$ are equi-probable.

Then, for duobinary and modified duobinary signals in the absence of noise, the received output levels have a (triangular) probability mass function of the form given by

$$p(b=2jd)=\frac{M-|j|}{M^2}, \qquad j=0,\pm 1,\pm 2,\cdots,\pm(M-1), \tag{8.4.25}$$

where b is the noise-free received level and

$2d$ is the distance between any two adjacent received signal levels.

The channel corrupts the transmitted signal by adding white Gaussian noise with zero-mean and power-spectral density $\frac{N_0}{2}$.

Assume that a symbol error is committed whenever the magnitude of the additive noise exceeds the distance d.

This assumption neglects the rare event that a large noise component with magnitude exceeding d may result in a received signal level that yields a correct symbol decision.

The noise component v_m is Gaussian with zero-mean and variance given by

$$\begin{aligned}\sigma_v^2 &= \frac{N_0}{2}\int_{-W}^{W} |G_R(f)|^2\, df \\ &= \frac{N_0}{2}\int_{-W}^{W} |X(f)|\, df \\ &= \frac{2N_0}{\pi}.\end{aligned} \tag{8.4.26}$$

for both the duobinary and the modified duobinary signals.

Hence, the symbol probability of error is upper-bounded as

$$\begin{aligned}P_M &< \sum_{j=-(M-2)}^{M-2} P(|y-2jd| > d \mid b=2jd)P(b=2jd) \\ &\qquad +2P\big(y+2(M-1)d > d \mid b=-2(M-1)d\big)P\big(b=-2(M-1)d\big) \\ &= P(|y|>d \mid b=0)\left[2\sum_{j=0}^{M-1} P(b=2jd) - P(b=0) - P\big(b=-2(M-1)d\big)\right] \\ &= \left(1-\frac{1}{M^2}\right)P(|y|>d \mid b=0).\end{aligned} \tag{8.4.27}$$

But

$$P(|y| > d \mid b = 0) = \frac{2}{\sqrt{2\pi}\sigma_v}\int_d^{\infty} e^{-\frac{x^2}{2\sigma_v^2}}dx$$

$$= 2Q\left(\sqrt{\frac{\pi d^2}{2N_0}}\right). \tag{8.4.28}$$

Therefore, the average probability of symbol error is upper-bounded as

$$P_M < 2\left(1 - \frac{1}{M^2}\right)Q\left(\sqrt{\frac{\pi d^2}{2N_0}}\right). \tag{8.4.29}$$

For the M-ary PAM signal with equi-probable transmitted level, the average power at the output of the transmitting filter is given by

$$P_{av} = \frac{E[a_m^2]}{T}\int_{-W}^{W}|G_T(f)|^2 df$$

$$= \frac{E[a_m^2]}{T}\int_{-W}^{W}|X(f)|\, df$$

$$= \frac{4}{\pi T}E[a_m^2] \tag{8.4.30}$$

where $E[a_m^2]$ is the mean square value of the M signal levels which is given by

$$E[a_m^2] = \frac{d^2(M^2-1)}{3}. \tag{8.4.31}$$

Therefore,

$$d^2 = \frac{3\pi P_{av} T}{4(M^2-1)}. \tag{8.4.32}$$

By plugging (8.4.32) into (8.4.29), the upper-bound for the symbol error probability is obtained as

$$P_M < 2\left(1-\frac{1}{M^2}\right) Q\left(\sqrt{\left(\frac{\pi}{4}\right)^2 \frac{6}{M^2-1} \frac{\varepsilon_{av}}{N_0}}\right) \tag{8.4.33}$$

where ε_{av} is the average energy/transmitted symbol, which can be also expressed in terms of the average bit energy by $\varepsilon_{av} = k\varepsilon_{bav} = (\log_2 M)\varepsilon_{bav}$.

(8.4.33) for the probability of error of M-ary PAM holds for both a duobinary and a modified duobinary partial response signal.

Comparing this result with the error probability of M-ary PAM with zero ISI which is obtained by using a signal pulse with a raised cosine spectrum, we note that the performance of partial response duobinary or modified duobinary has a loss of $\left(\frac{\pi}{4}\right)^2$ or 2.1 dB.

This loss in SNR is due to the fact that the detector for the partial response signals makes decisions on a symbol-by-symbol basis, thus, ignoring the inherent memory contained in the received signal at the input to the detector.

To observe the memory in the received sequence, look at the noise-free received sequence for binary transmission given in Table 8.1.

Table 8.1 Binary Signaling with Duobinary Pulses.

Data sequence d_n		1	1	1	0	1	0	0	1	0	0	0	1	1	0	1
Precoded sequence p_n	0	1	0	1	1	0	0	0	1	1	1	1	0	1	1	0
Transmitted sequence a_n	−1	1	−1	1	1	−1	−1	−1	1	1	1	1	−1	1	1	−1
Received sequence b_n		0	0	0	2	0	−2	−2	0	2	2	2	0	0	2	0
Decoded sequence d_n		1	1	1	0	1	0	0	1	0	0	0	1	1	0	1

Note that it is not possible to have a transition from -2 to 2 or vice versa between two successive received samples from the matched filter.

However, a symbol-by-symbol detector does not exploit this constraint or inherent memory in the received sequence.

8.5 Digitally Modulated Signals with Memory

The spectrum shaping obtained with partial response signals (in Section 8.3) can be viewed as resulting from the memory introduced in the transmitted sequence of symbols, that is, the sequence of symbols are correlated and, as a consequence, the power spectral density of the transmitted symbol sequence is colored (nonwhite).

Signal dependence among signals transmitted in different signal intervals is generally accomplished by encoding the data at the input to the modulator by means of a **modulation code**.

8.5.1 Modulation Codes and Modulation Signals with Memory

Modulation codes are usually employed in magnetic recording, optical recording, and digital communications over cable systems to achieve spectral shaping of the modulated signal that matches the passband characteristics of the channel.

In magnetic recording we encounter two bais problems: 1) the packing density and 2) the dc content.

The first problem in magnetic recording is that we need to write as many bits as possible on a single track of magnetic medium (disk or tape) to ahieve high packing density.

However, the medium imposes a limit on how closely successive bits in a sequence are stored.

Figure 8.15 shows a block diagram of the magnetic recording system.

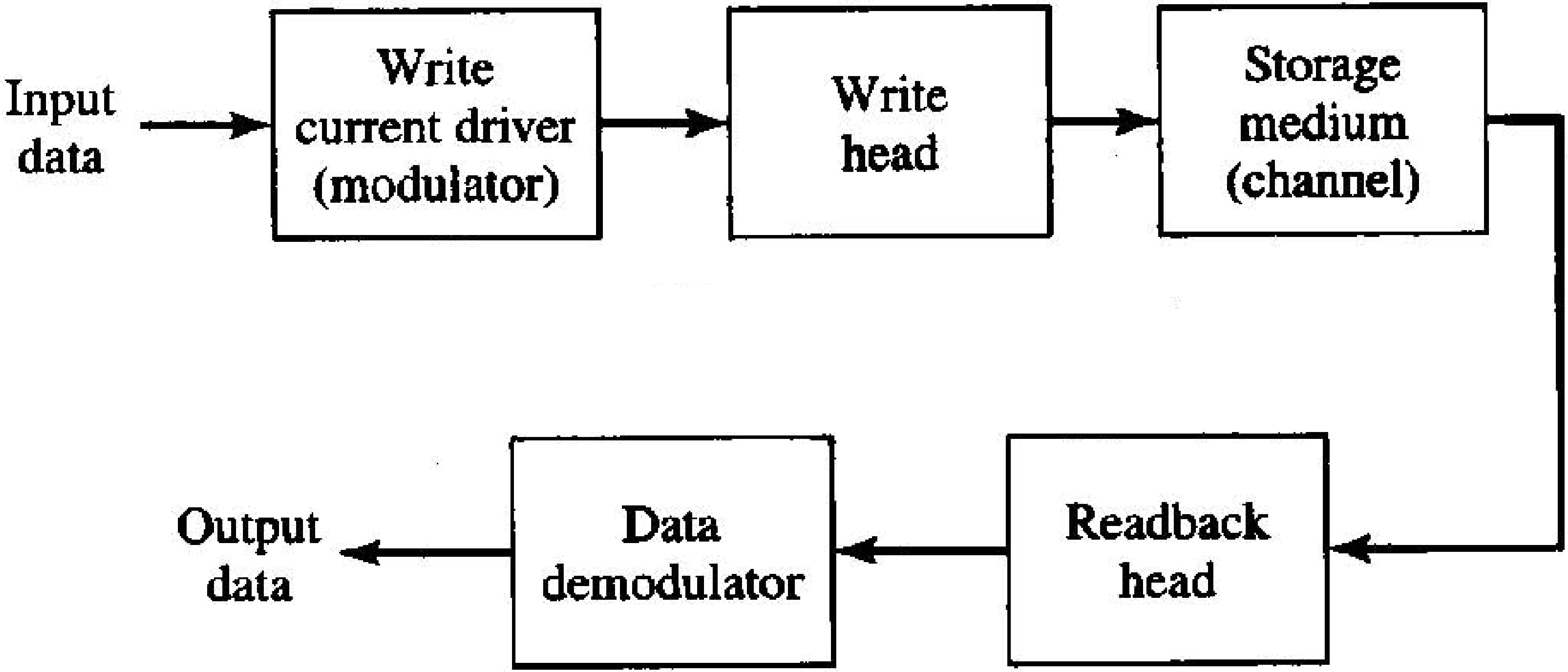


Figure 8.15 Block diagram of magnetic storage read/write system.

The binary input data sequence is used to generate a write current which can be viewed as the output of the "modulator."

The two most commonly used methods to map the data sequence into the write current waveform are the NRZ (non-return-to-zero) and NRZI (non-return-to-zero inverse) methods of which waveforms are shown in Figure 8.16.

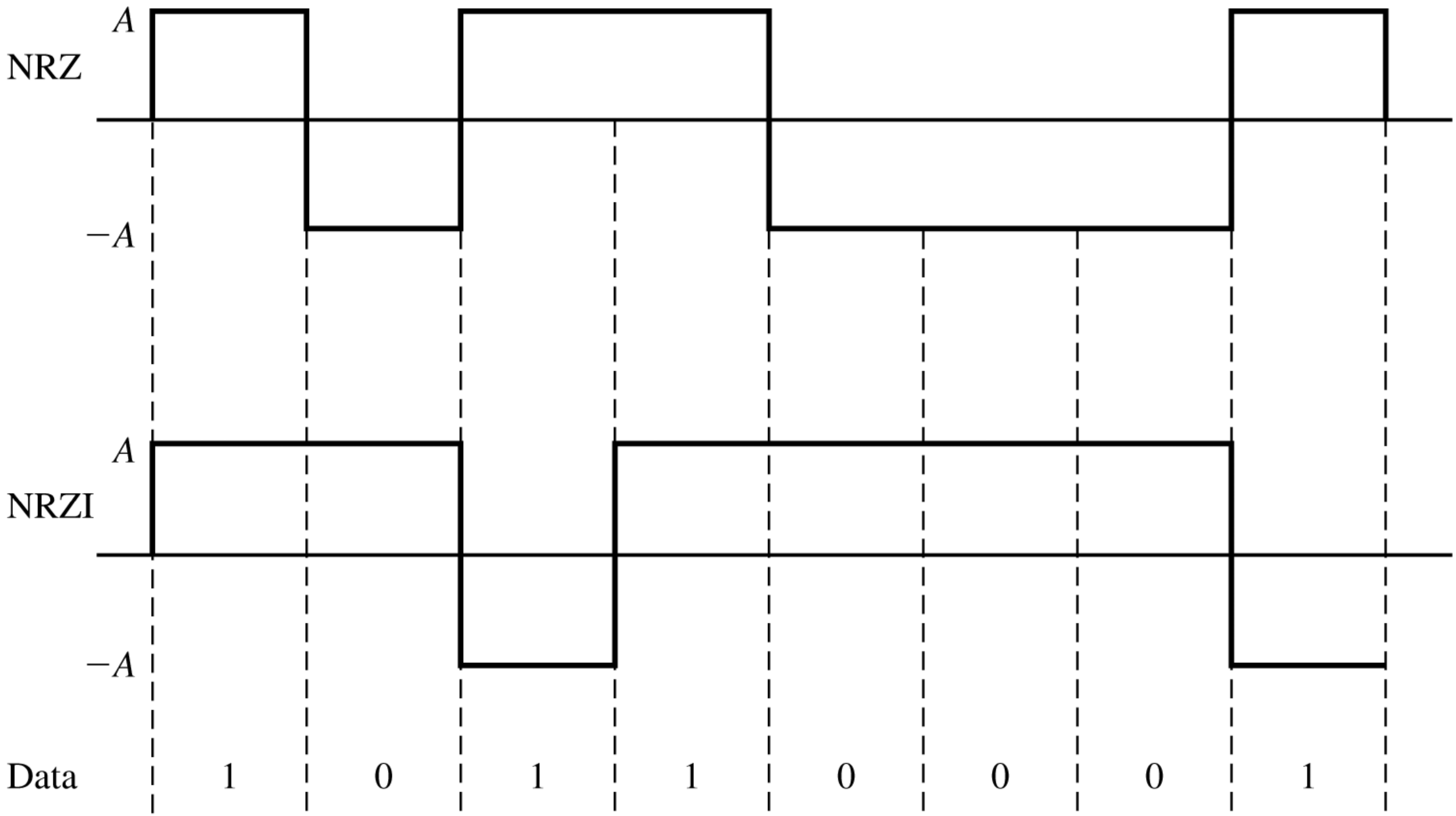


Figure 8.16

NRZ and NRZI signals.

Note that NRZ is identical to binary PAM in which the information bit 1 is represented by a rectangular pulse of amplitude A and the information bit 0 is represented by a rectangular pulse of amplitude $-A$.

In the NRZI, transitions from one amplitude level to another (A to $-A$, or $-A$ to A) occur only when the information bit is a 1.

In the NRZI, no transition occurs when the information bit is a 0, that is, the amplitude level remains the same as the previous signal level.

The positive amplitude pulse results in magnetizing the medium on one (direction) polarity and the negative pulse magnetizes the medium in the opposite (direction) polarity.

Assume that the input data sequence is the sequence of independent random variables with equi-probable 1's and 0's.

Then, whether we use NRZ or NRZI, there are level transitions for A to $-A$ or $-A$ to A with probability $\frac{1}{2}$ (that is, each with probability $\frac{1}{4}$) for every data bit.

Suppose that the readback signal for a positive transition ($-A$ to A) is a pulse modeled as

$$p(t) = \frac{1}{1+\left(\frac{2t}{T_{50}}\right)^2} \tag{8.5.1}$$

where T_{50} is the width of the pulse at its 50% amplitude level as shown in Figure 8.17.

The value of T_{50} is determined by the characteristics of the medium and the read/write heads.

Then, the readback signal for a negative transition (A to $-A$) becomes the pulse $-p(t)$.

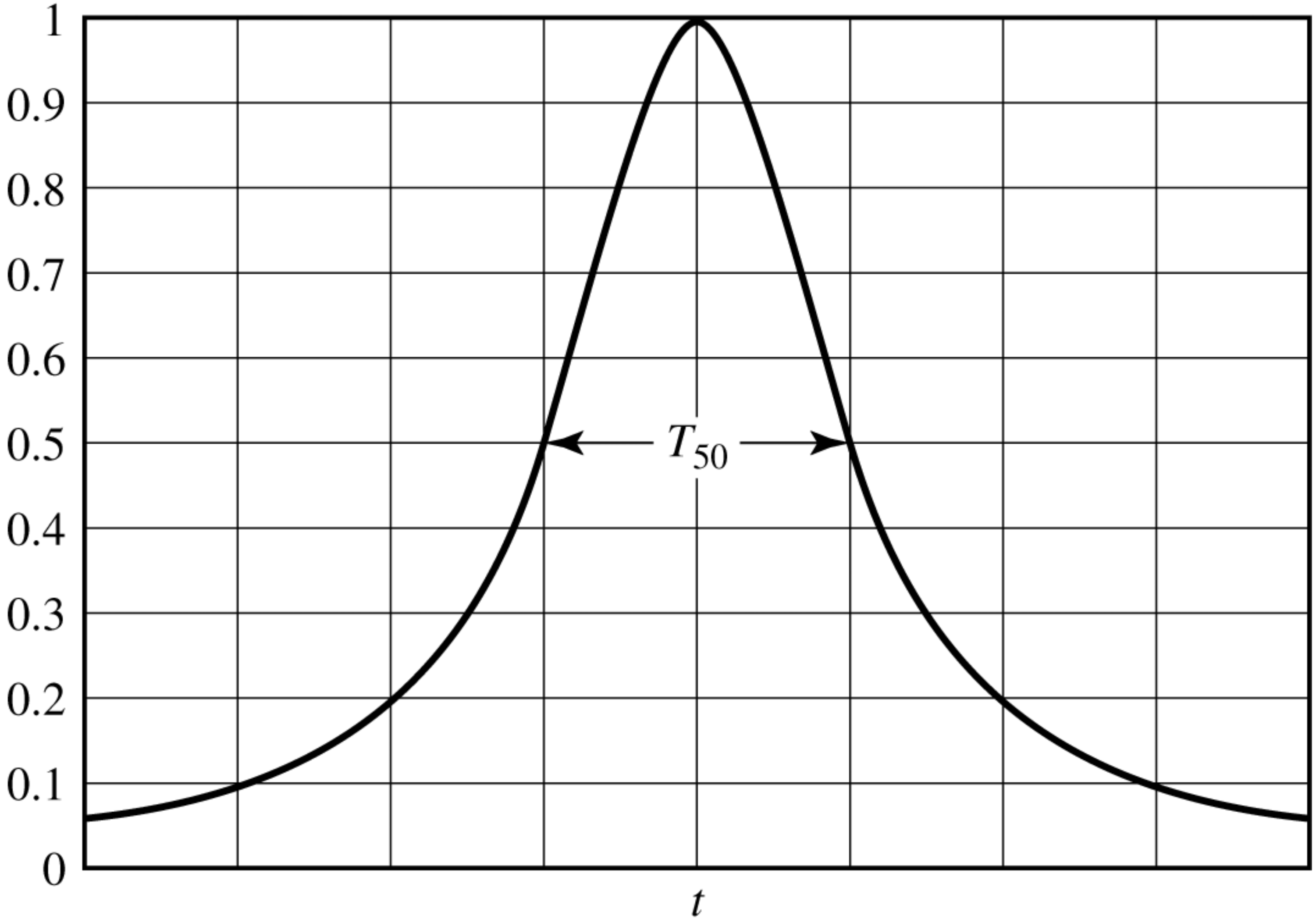


Figure 8.17

Readback pulse in magnetic recording system.

Consider the case that we write a positive transition followed by a negative transition.

The readback signal pulses are obtained by superposing $p(t)$ with $-p(t-T_b)$ where T_b is the bit time interval, that is, the time interval between two transitions.

The closer the bit transitions (T_b small) are, the larger will be the normalized density and, hence, the larger will be the packing density.

Figure 8.18 shows the readback signal pulses for various values of the **normalized density** $\Delta = \frac{T_{50}}{T_b}$.

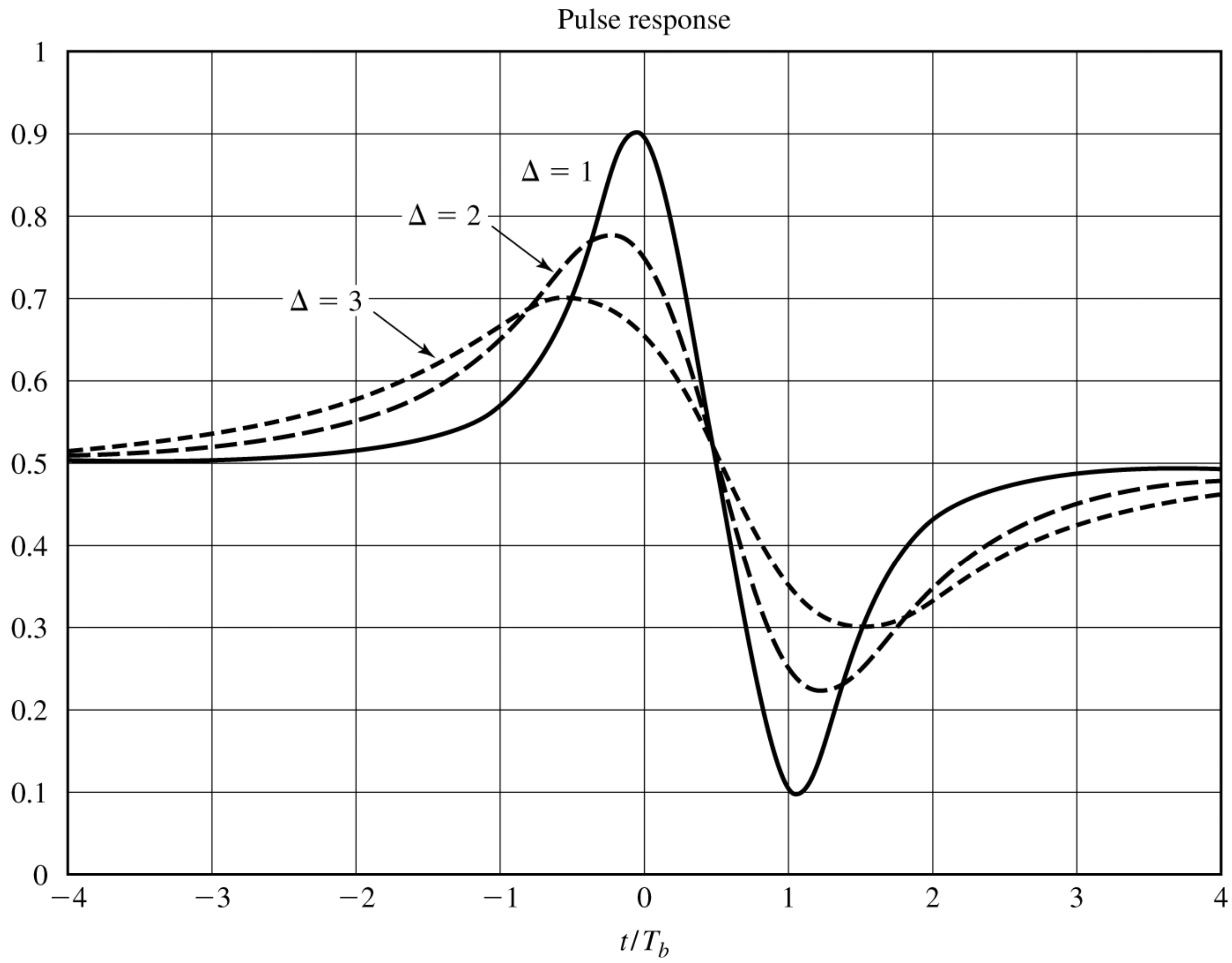


Figure 8.18

Readback signal response to a pulse.

Notice that as the normalized density Δ is increased, the peak amplitudes of the readback signal are reduced and are shifted in time from the desired time instants.

This implies that the pulses interfere with each another, limiting the density with which we can write transitions.

It needs to design modulation codes that take the original data sequence and transform (or encode) it into another sequence which results in a write waveform having amplitude transitions spaced further apart.

For example, if we use NRZI, the encoded sequence into the modulator contains one or more 0's between 1's.

The second problem in magnetic recording is that we want to avoid (or minimize) having a dc component in the modulated signal (the write current) because of the frequency-response characteristics of the readback system and digital communication over cable channels.

This problem can also be overcome by altering (encoding) the data sequence to the modulator.

A class of codes that solves these two problems are **modulation codes**.

Runlength-Limited Codes

Codes that have a restriction on the number of consecutive 1's or 0's in a sequence are generally called **runlength-limited codes**.

These codes are generally described by two parameters: d and κ ('kappa'),
where d is the minimum number of 0's between 1's in a sequence, and
κ is the maximum number of 0's between two 1's in a sequence.

When used with NRZI modulation, placing d zeros between successive 1's spreads the transitions farther apart, thus, reducing the overlap in the channel response due to successive transitions.

Setting an upper limit κ on the runlength of 0's ensures that transitions occur frequently enough so that symbol timing information can be recovered from the received modulated signal.

Runlength-limited codes are usually called (d, κ) codes.†

The (d, κ) code sequence constraints may be represented by a finite-state sequential machine with $\kappa + 1$ states which is denoted by σ_i, $1 \le i \le \kappa + 1$, as shown in Figure 8.19.

† Runlength-limited codes are usually called (d, k) codes, where k is the maximum runlength of zeros. We have substituted the Greek letter κ for k, to avoid confusion with our previous use of k.

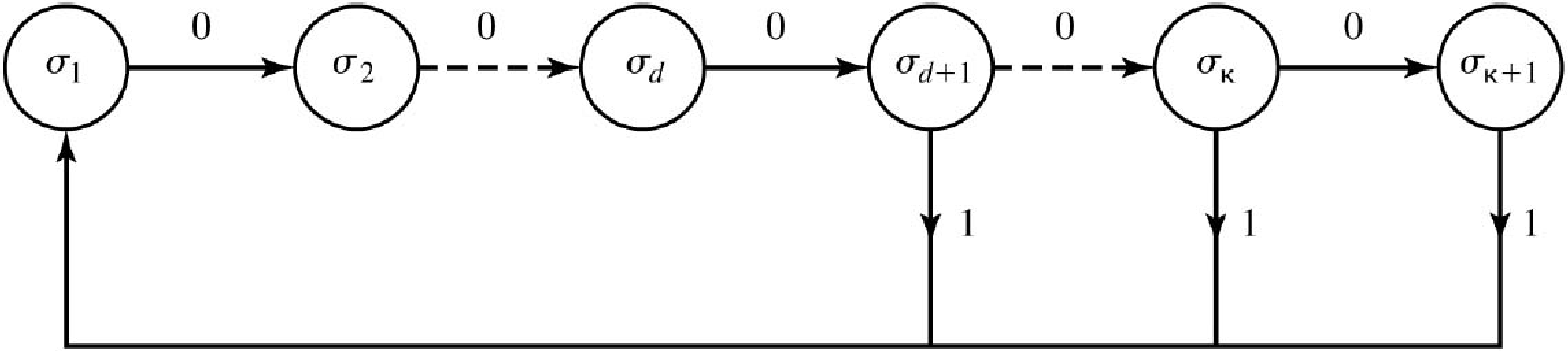


Figure 8.19

Finite-state sequential machine for a (d, κ) -coded sequence.

We observe that an output data bit 0 takes the sequence from state σ_i to σ_{i+1}, $i \le \kappa$.

The output data bit 1 takes the sequence to state σ_1.

The output bit from the encoder may be 1 only when the sequence is in state σ_i, $d+1 \le i \le \kappa+1$.

When the sequence is in state $\sigma_{\kappa+1}$, the output bit is always 1.

The finite-state sequential machine is also represented by a **state transition matrix** **D** which is a $(\kappa+1)\times(\kappa+1)$ square matrix of which elements $d_{i\,j}$ are given by

$$d_{i\,1} = 1, \quad i \geq d+1,$$

$$d_{i\,j} = \begin{cases} 1, & j = i+1, \\ 0, & \text{otherwise.} \end{cases} \tag{8.5.2}$$

Ex. 8.5.1

Determine the state transition matrix for a $(d,\kappa)=(1,3)$ code.

Solution

The $(1,3)$ code has $(\kappa+1=)$ 4 states.

From Figure 8.19 the state transition matrix of the code is obtained as

$$\mathbf{D} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}. \tag{8.5.3}$$

Let $N(n)$ denotes the number of sequences of length n which satisfy the (d, κ) constraints.

Then, as n is allowed to increase, the number of sequences $N(n)$ also increases.

The number of information bits that can be uniquely represented with $N(n)$ code sequences is given by

$$k = \lfloor \log_2 N(n) \rfloor$$

where $\lfloor x \rfloor$ denotes the largest integer contained in x.

Then maximum code rate is given by

$$R_c = \frac{k}{n}.$$

The capacity of a (d, κ) code is defined as

$$C(d, \kappa) = \lim_{n\to\infty} \frac{1}{n} \log_2 N(n) . \tag{8.5.4}$$

which is the maximum possible rate that can be achieved with the (d, κ) constraints.

Shannon (1948) showed that the capacity the (d, κ) code is given by

$$C(d,\kappa) = \log_2 \lambda_{\max} \tag{8.5.5}$$

where $\lambda_{\max}$ is the largest real eigenvalue of the state transition matrix $\mathbf{D}$ the code.

Ex. 8.5.2

Determine the capacity of a $(d, \kappa) = (1, 3)$ code.

Solution

Using the state-transition matrix given in Example 8.5.1 for the $(1, 3)$ code, we have

$$\det(\mathbf{D}-\lambda\mathbf{I}) = \det\begin{bmatrix} -\lambda & 1 & 0 & 0 \\ 1 & -\lambda & 1 & 0 \\ 1 & 0 & -\lambda & 1 \\ 1 & 0 & 0 & -\lambda \end{bmatrix}$$

$$= \lambda^4 - \lambda^2 - \lambda - 1$$

$$= 0.$$

The maximum real root of this equation is $\lambda_{\max} = 1.4656$.

Therefore, the capacity of the code is given by

$$C(1,3) = \log_2 \lambda_{\max}$$

$$= 0.5515.$$

The capacities of (d, κ) codes for $0 \le d \le 6$ and $2 \le \kappa \le 15$ are shown in Table 8.3.

Table 8.3 Capacity $C(d,\kappa)$ versus runlength parameters d and κ.

κ	$d=0$	$d=1$	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$
2	.8791	.4057					
3	.9468	.5515	.2878				
4	.9752	.6174	.4057	.2232			
5	.9881	.6509	.4650	.3218	.1823		
6	.9942	.6690	.4979	.3746	.2269	.1542	
7	.9971	.6793	.5174	.4057	.3142	.2281	.1335
8	.9986	.6853	.5293	.4251	.3432	.2709	.1993
9	.9993	.6888	.5369	.4376	.3620	.2979	.2382
10	.9996	.6909	.5418	.4460	.3746	.3158	.2633
11	.9998	.6922	.5450	.4516	.3833	.3285	.2804
12	.9999	.6930	.5471	.4555	.3894	.3369	.2924
12	.9999	.6935	.5485	.4583	.3937	.3432	.3011
14	.9999	.6938	.5495	.4602	.3968	.3478	.3074
15	.9999	.6939	.5501	.4615	.3991	.3513	.3122
∞	1.000	.6942	.5515	.4650	.4057	.3620	.3282

In general, (d, κ) codes can be constructed either as fixed-length codes or variable-length codes.

From Table 8.3, we observe that the capacity $C(d,\kappa) < \frac{1}{2}$ for any value of κ when $d \geq 3$.

The most commonly used codes for magnetic recording have the minimum number of 0's between 1's in a sequence $d \leq 2$.

In a fixed-length code, each bit or block of k bits is encoded into a block of n $(>k)$ bits.

To construct a fixed-length code with a given block length n, we may select the subset of the 2^n codewords that satisfy the specified runlength constraints.

From this subset, we eliminate codewords that do not satisfy the runlength constraints when they are concatenated.

Thus, we obtain a set of codewords that satisfies the runlength constraints which can be used in the mapping of the input data bit sequences to the encoder.

The encoding and decoding can be performed by using a look-up table.

Ex. 8.5.3

Construct a $d = 0$, $\kappa = 2$ code of length $n = 3$, and determine its efficiency.

Solution

By listing all possible words, we find that the following five codewords which satisfy the $(0, 2)$ constraint:

$(0\,1\,0)$, $(0\,1\,1)$, $(1\,0\,1)$, $(1\,1\,0)$, $(1\,1\,1)$.

We may select any four of these five codewords and use them to encode the data sequences of two bits $(0\,0, 0\,1, 1\,0, 1\,1)$.

Thus, we have a rate $R_c = \frac{k}{n} = \frac{2}{3}$ code that satisfies the $(0, 2)$ constraint.

The capacity of a $(0, 2)$ code is given by

,

$$
\begin{aligned}
C(0,2) &= \lim_{n\to\infty} \frac{1}{n} \log_2 N(n) \\
&= \frac{1}{3} \cdot \log_2 4 \\
&= 0.8791,
\end{aligned}
$$

so that this code has an **efficiency** of

$$
\begin{aligned}
\text{efficiency} &= \frac{R_c}{C(d,\kappa)} \\
&= \frac{\frac{2}{3}}{0.8791} \\
&= 0.76.
\end{aligned}
$$

Note that the fixed-length code in this example is not very efficient.

Better $(0,2)$ codes can be constructed by increasing the block length n.

Ex. 8.5.4

Construct a $d=1,\ \kappa=\infty$ code of length $n=5$.

Solution

From the set of $(2^5=)\ 32$ possible words, we select the codewords which satisfy the constraint of $d=1$.

There are eight such codewords so that we can encode three information bits with each codeword.

The code is given in Table 8.4.

Note that the first bit of each codeword is 0, whereas the last bit may be either 0 or 1, so that the $d=1$ constraint is satisfied when these codewords are concatenated.

This code has a rate $R_c=\frac{3}{5}$.

From Table 8.3, the capacity is obtained as $C(1,\infty)=0.6942$.

Hence, the code efficiency is given by $\text{efficiency}=\frac{R_c}{C(d,\kappa)}=\frac{\frac{3}{5}}{0.6942}=0.864$, which is quite acceptable.

Table 8.4 Fixed length $d = 1,\ \kappa = \infty$ code.

Input data bits	Output coded sequence
000	00000
001	00001
010	00010
011	00100
100	00101
101	01000
110	01001
111	01010

The code construction in the two examples above produces fixed-length (d,κ) codes that are **state independent** which means that fixed-length codewords can be concatenated without violating the (d,κ) constraints.

In general, fixed-length, state-independent (d,κ) codes require large block lengths, except in cases such as those in the examples above, where d is small.

Simpler (shorter-length) codes are generally possible by allowing for state-dependence and for variable-length codewords.

Ex. 8.5.5

Construct a simple variable-length $d=0$, $\kappa=2$ code.

Solution

A very simple uniquely decodable $(0,2)$ code is as follows:

$$\begin{aligned} 0 &\rightarrow 01 \\ 10 &\rightarrow 10 \\ 11 &\rightarrow 11. \end{aligned}$$

This code has a fixed output block length but a variable input block size.

In general, both the input and output blockelngth may be variable. This case is shown in the following example.

Ex. 8.5.6

Construct a $(2, 7)$ variable blocklength code.

Solution

The $(2, 7)$ code has been widely used by IBM in many of its disk storage systems.

The code is given in Table 8.5.

Table 8.5 Coded book for variable-length $(2, 7)$ code.

Input data bits	Output coded sequence
10	1000
11	0100
011	000100
010	001000
000	100100
0011	00100100
0010	00001000

We observe that the input data blocks of 2, 3, and 4 bits are mapped into output data blocks of 4, 6, and 8 bits, respectively.

Hence, the code rate is given by

$$R_c = \frac{4}{8}$$
$$= \frac{1}{2}.$$

Since this is the code rate for all codewords, the code is called a **fixed-rate** code.

This code has an efficiency $= \dfrac{0.5}{0.5174} = 0.966$.

Note that this code satisfies the prefix condition.

Another code that has been widely used in magnetic recording is the rate $\frac{1}{2}$, $(d, \kappa) = (1, 3)$ code given in Table 8.6.

Table 8.6 Encoder for $(1, 3)$ Miller code.

Input data bits	Output coded sequence
0	$x0$
1	01

We observe that when the information bit is 0, the first output bit is either 1, if the previous input bit was 0, or 0, if the previous input bit was 1.

When the information bit is 1, the encoder output is $0\,1$.

Decoding of this code is simple.

The first bit of the two-bit block is redundant and may be discarded.

The second bit is the information bit.

This code is called the **Miller code**.

This is a **state-dependent code** which is described by the state diagram with two states S_1 and S_2 shown in Figure 8.20.

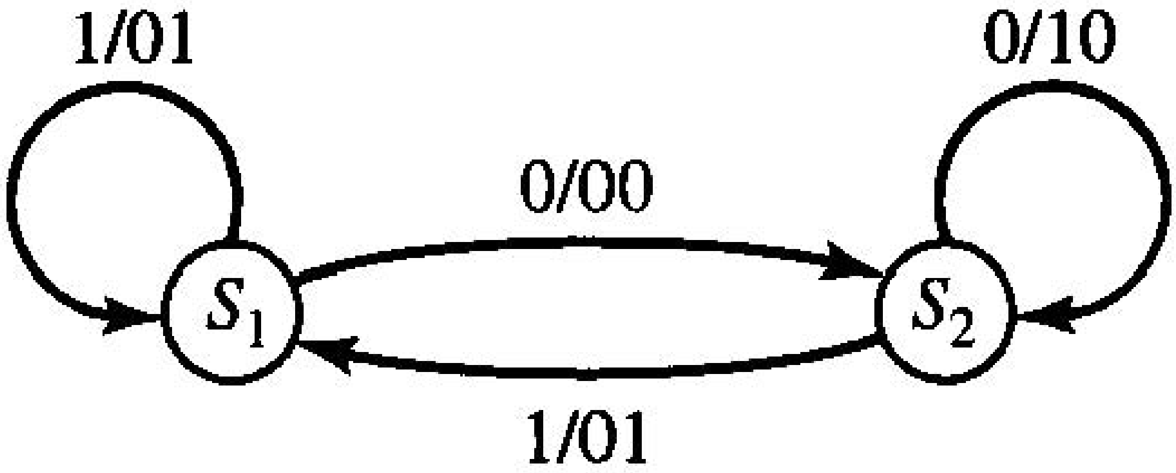


Figure 8.20 State diagrams for $d = 1$, $\kappa = 3$ (Miller) code.

When the encoder is a state S_1, an input bit 1 results in the encoder staying in state S_1 and outputs 01. This is represented by $1/01$.

Trellis Representation of State-Dependent (d,κ) Codes

Another way to describe codes that have memory is by means of a **trellis** which is a graph showing the state transitions as a function of time.

A trellis consists of a set of nodes representing the states which characterize the memory in the code at different time instants and interconnections between pairs of nodes which indicate the transitions between successive time instants.

Figure 8.21 shows the trellis for the $d=1$, $\kappa=3$ Miller code whose state diagram is shown in Figure 8.20.

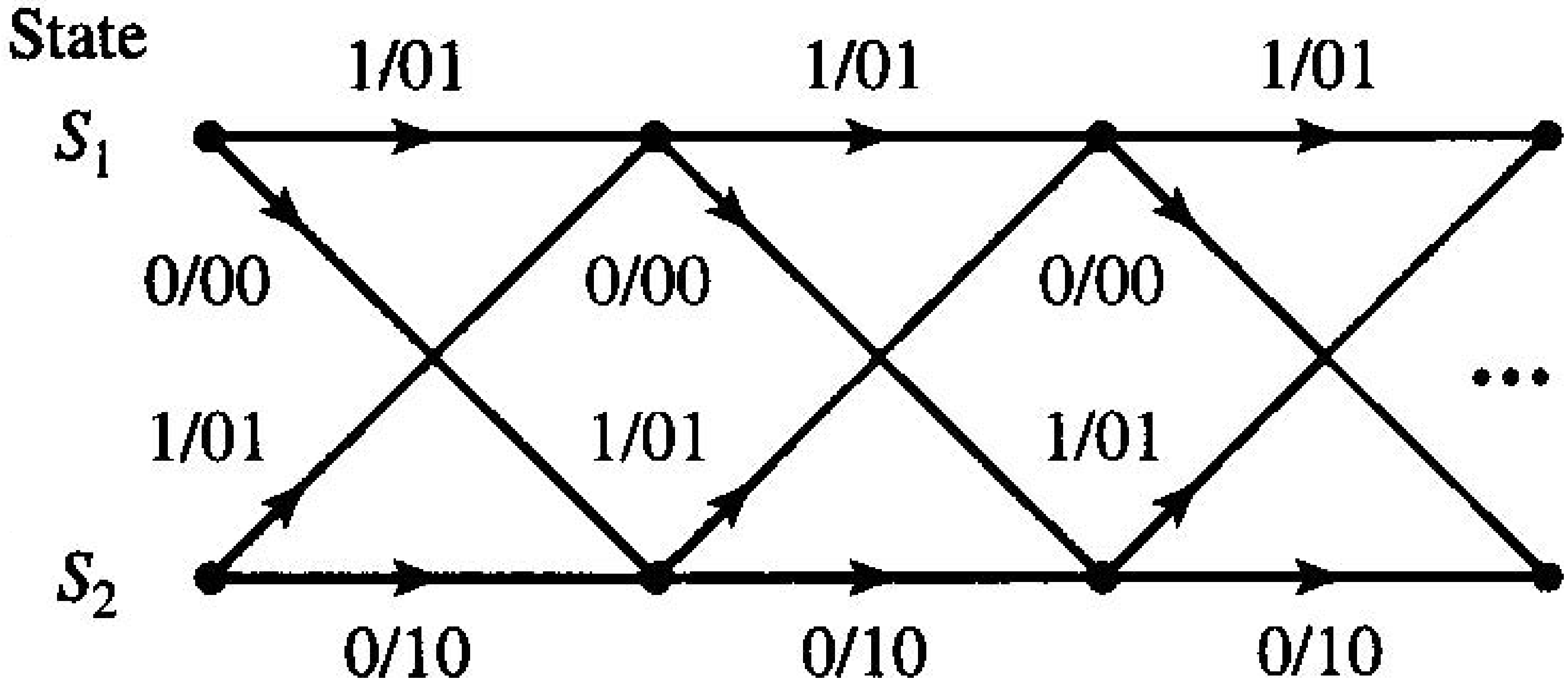


Figure 8.21 Trellis for $d = 1$, $\kappa = 3$ (Miller) code.

Mapping of Coded Bits into Signal Waveforms

The output sequence from the (d,κ) encoder is mapped by the modulator into signal waveforms for transmission over the channel.

If the binary digit 1 is mapped into a rectangular pulse of amplitude A and the binary digit 0 is mapped into a rectangular pulse of amplitude $-A$, the result is a (d,κ)-coded NRZ modulated signal.

Note that the duration of the rectangular pulses is $T_c = \frac{R_c}{R_b} = R_c T_b$,

where R_b is the information (bit) rate into the encoder,

T_b is the corresponding (uncoded) bit interval, and

R_c is the code rate for the (d,κ) code.

When the (d,κ) code is a state-independent fixed-length code with code rate $R_c = \frac{k}{n}$, we may consider each n-bit block as generating one signal waveform of duration nT_c.

We have $M = 2^k$ signal waveforms, one for each of the 2^k possible k-bit data blocks. These coded waveforms are given by (7.4.22) and (7.4.23).

In this case, there is no dependence between the transmission of successive waveforms.

The modulation signal is no longer memoryless when NRZI is used and, or, the (d, κ) code is state dependant.

First, consider the effect of mapping the coded bits into an NRZI signal waveform.

An NRZI modulated signal is itself state dependent.

The signal-amplitude level is changed from its current value $(\pm A)$ only when the bit to be transmitted is 1.

Note that the NRZI signal may be viewed as an NRZ signal preceded by another encoding operation, called **precoding**, of the binary sequence, as shown in Figure 8.22.

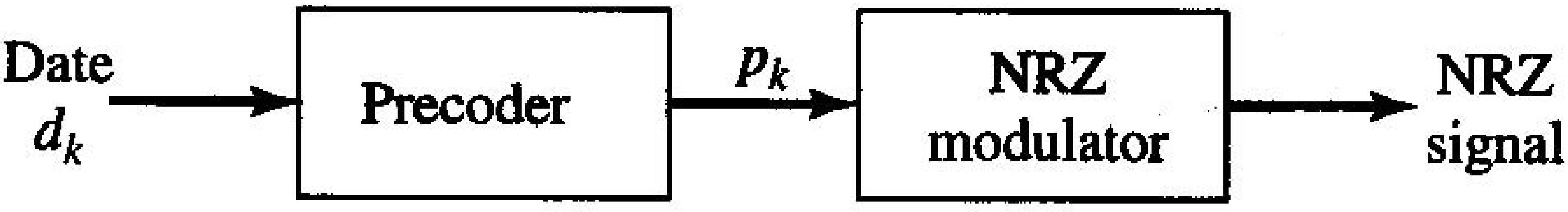


Figure 8.22 Method for generating an NRZI signal using precoding.

The precoding operation is described mathematically by the relation

$$p_k = d_k \oplus p_{k-1}$$

where $\{d_k\}$ is the binary sequence into the precoder,

$\{p_k\}$ is the output binary sequence from the precoder, and

$\oplus$ denotes modulo-2 addition.

This type of encoding is called **differential encoding**, and is characterized by the atate diagram shown in Figure 8.23(a).

Then, the sequence $\{p_k\}$ is transmitted by NRZ.

When $p_k = 1$, the modulator output is a rectangular pulse of amplitude A and when $p_k = 0$, the modulator output is a rectangular pulse of amplitude $-A$.

When the signal waveforms are superimposed on the state diagram of Figure 8.23(a), we obtain the corresponding state diagram shown in Figure 8.23(b).

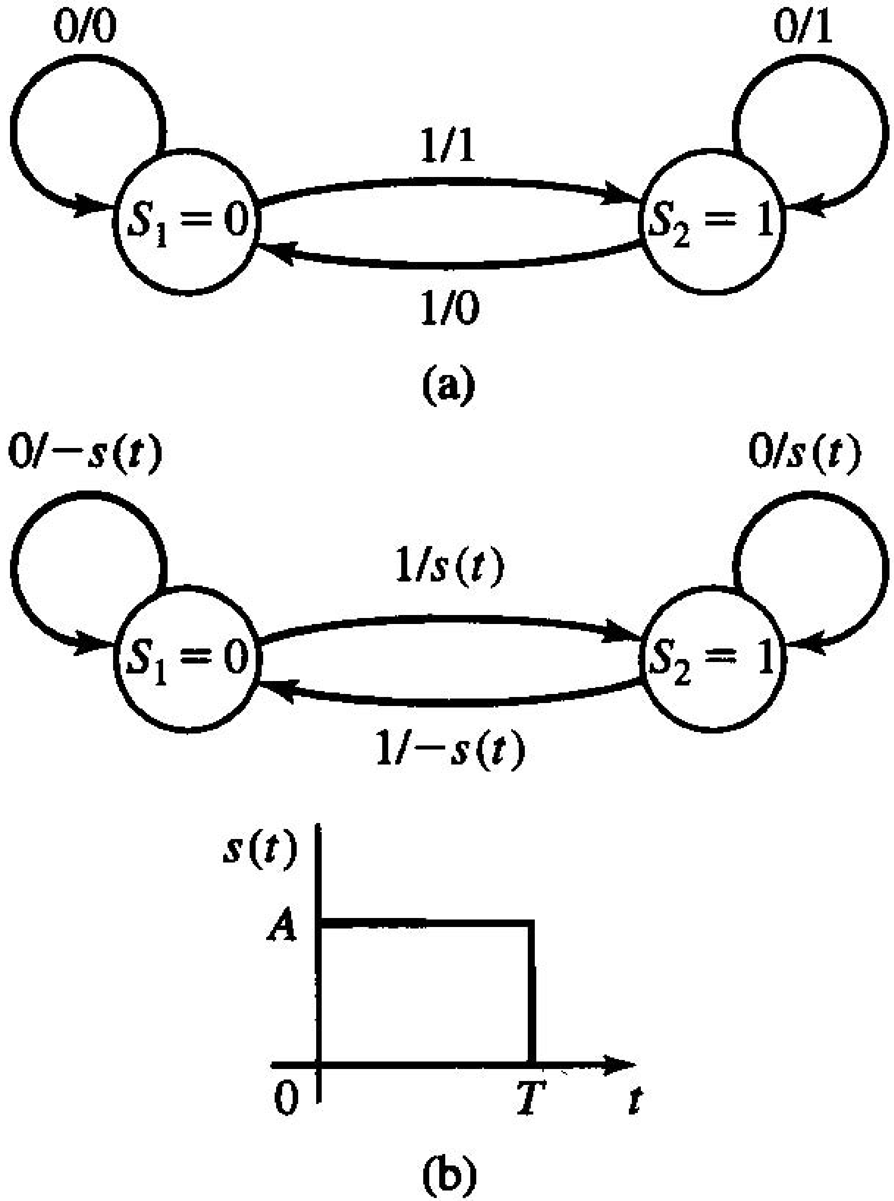


Figure 8.23 State diagram for NRZI signal.

Differential encoding or precoding as described above introduces memory in the modulated signal.

As in the case of state-dependent (d, κ) codes, a trellis diagram may be used to show the time dependence of the modulated signal.

The trellis diagram for the NRZI signal is shown in Figure 8.24.

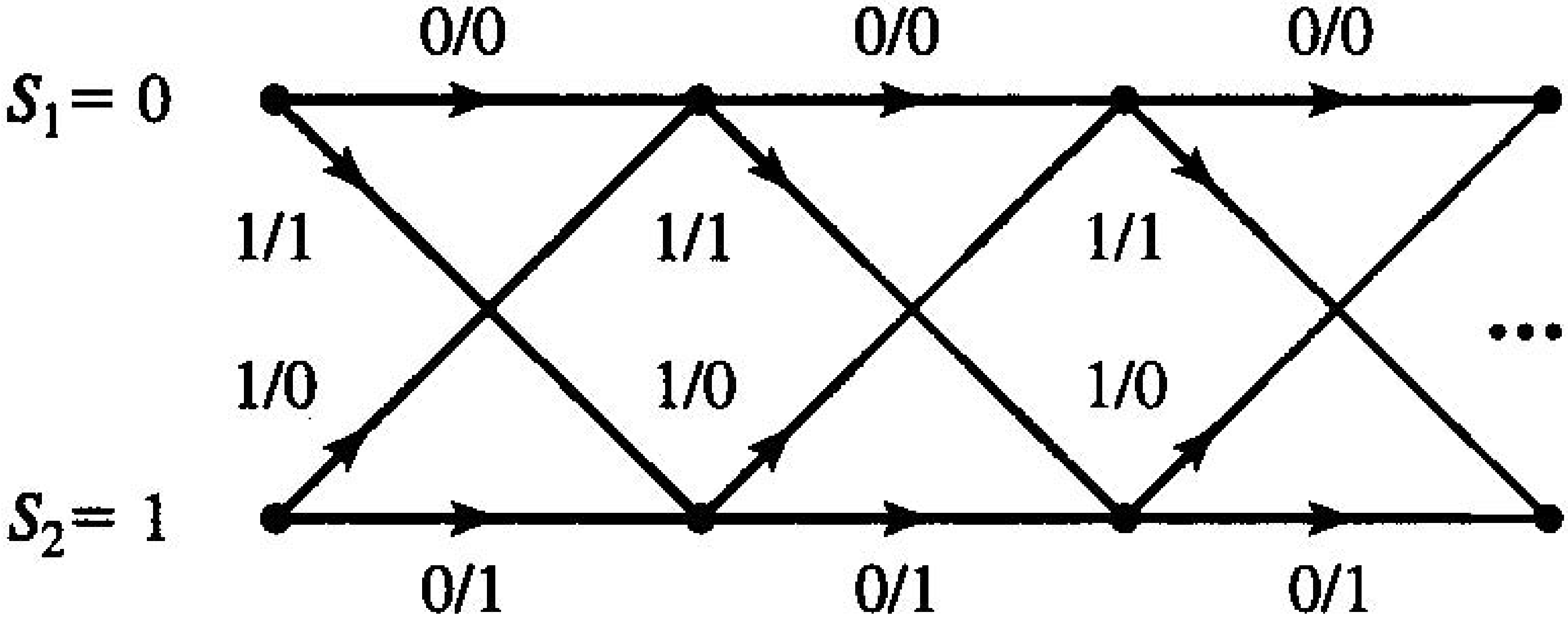


Figure 8.24 Trellis for an NRZI signal.

When the output of a state-dependent (d, κ) encoder is followed by an NRZI modulator, we may simply combine the two state diagrams into a single state diagram for the (d, κ) code with precoding.

A similar combination can be performed with the corresponding trellises.

Ex. 8.5.7

Determine the state diagram of the combined $(1,3)$ Miller code followed by the precoding inherent in NRZI modulation.

Solution

Since the $(1,3)$ Miller code has two states and the precoder has two states, the state diagram for the combined encoder has four states, which are denoted as $(S_M, S_N) = (\sigma_1, s_1), (\sigma_1, s_2), (\sigma_2, s_1), (\sigma_2, s_2)$,

where $S_M = \{\sigma_1, \sigma_2\}$ represents the two states of the Miller code and

$S_N = \{s_1, s_2\}$ represents the two states of the precoder for NRZI.

For each data input bit into the Miller encoder we obtain two output bits which are then precoded to yield two precoded output bits.

The resulting state diagram is shown in Figure 8.25, where the first bit denotes the information bit into the Miller encoder and the next two bits represent the corresponding output of the recoder.

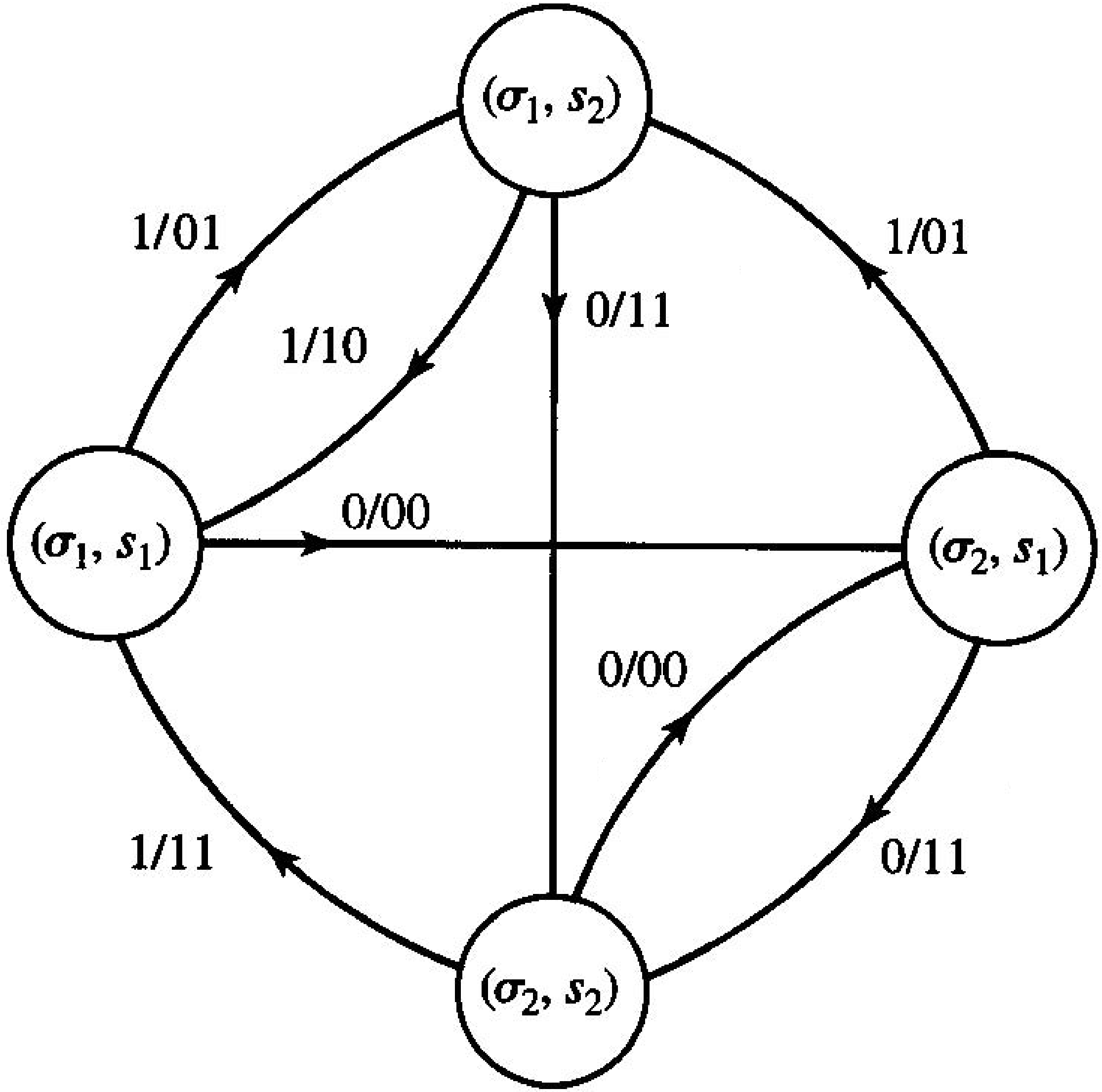


Figure 8.25 State diagram of the Miller code followed by the precoder.

The trellis diagram for the Miller precoded sequence may be obtained directly from the combined state diagram or from a combination of the trellises of the two codes.

The result of this combination is the four-state trellis, one stage of which is shown in Figure 8.26.

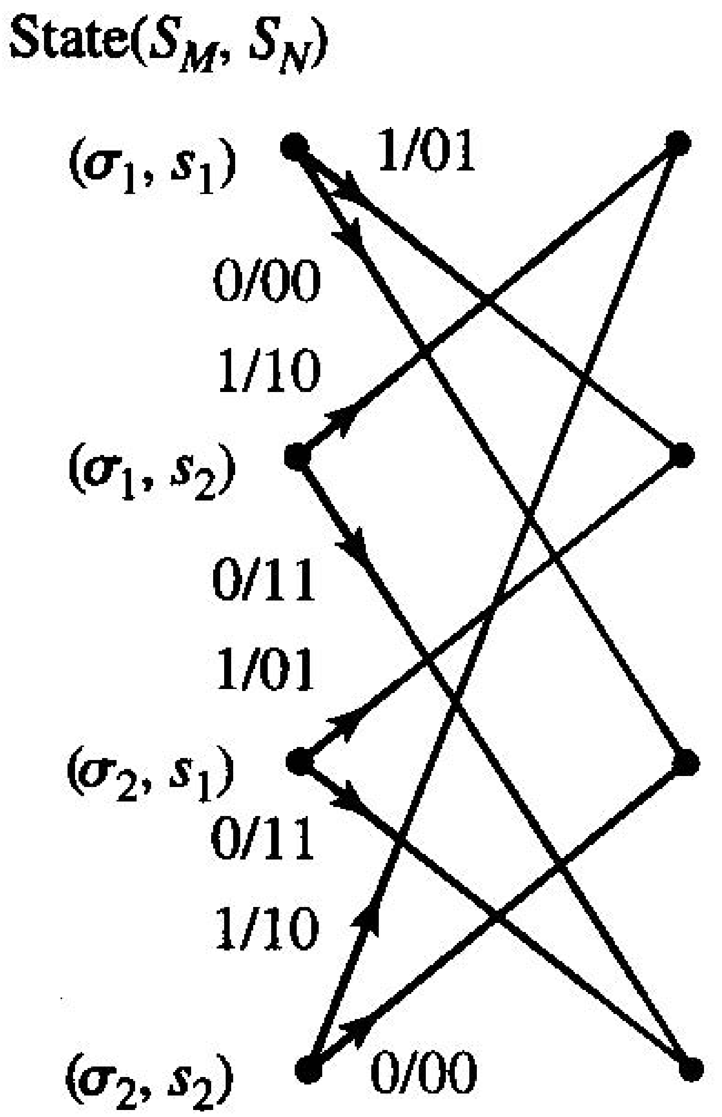
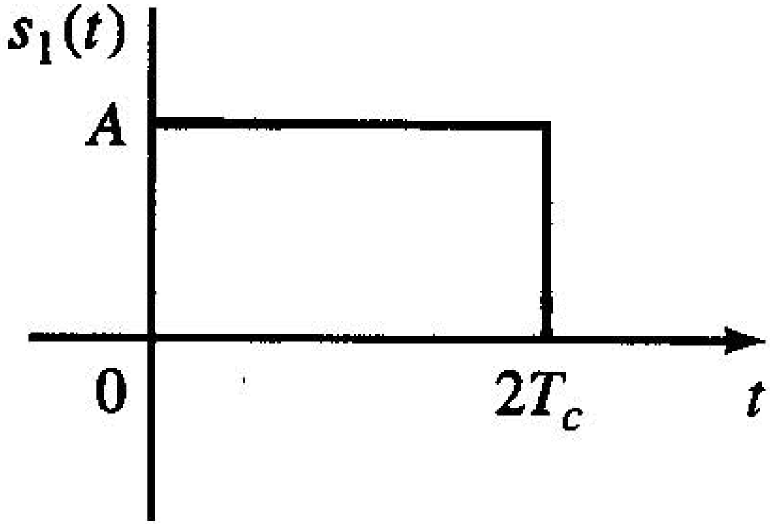


Figure 8.26 One stage of trellis diagram for the Miller code followed by the precoder.

Note that the four signal waveforms obtained by mapping each pair of bits of the Miller-precoded sequence into an NRZ signal are biorthogonal.

In particular, the pair of bits 11 map into the waveform $s_1(t)$ and the bits 01 map into the waveform $s_2(t)$, shown in Figure 8.27.



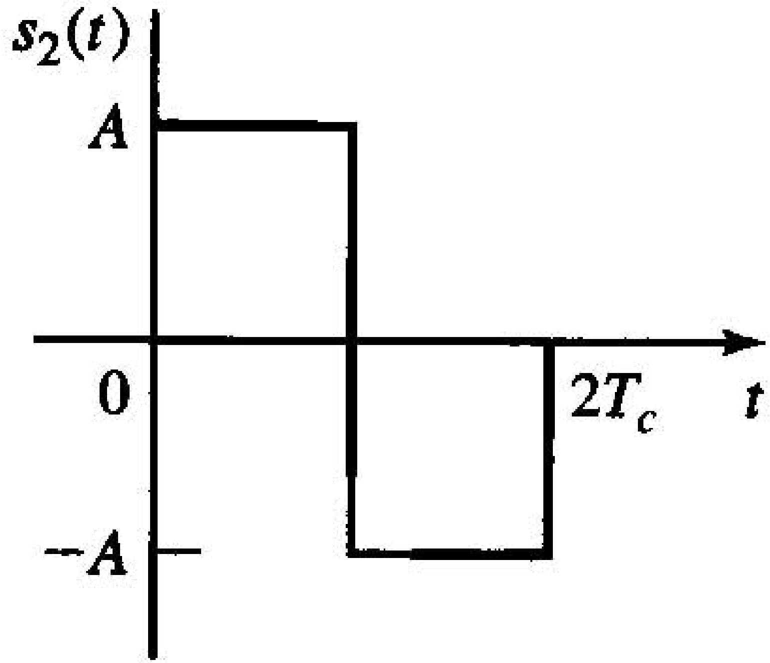
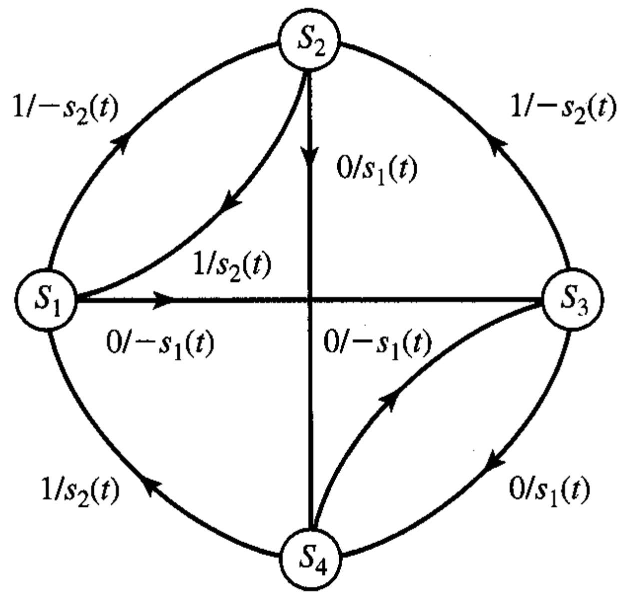


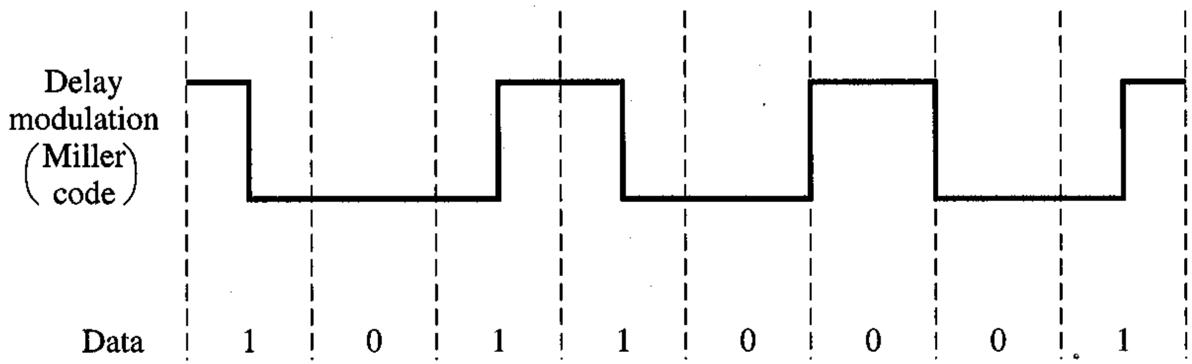
Figure 8.27 Signal waveforms for Miller-precoded pairs of bits.

Then, the encoded bits $0\,0$ map into $-s_1(t)$ and the bits 10 map into $-s_2(t)$.

Since $s_1(t)$ and $s_2(t)$ are orthogonal, the set of four waveforms constitute a biorthogonal set of $M = 4$ waveforms.



(a)



(b)

Figure 8.28 State diagram for the Miller-precoded signal and sample waveform.

The state diagram is shown in Figure 8.28, where the four states are simply designated as S_i, $1 \le i \le 4$.

The resulting modulated signal has also been called **delay modulation**.

Modulated signals with memory such as NRZI and Miller coded/NRZI (delay modulation) are generally characterized by a K-state Markov chain with **stationary state probabilities** $\{p_i,\ i = 1, 2, \cdots, K\}$.

The transition probability p_{ij} denotes the probability that signal waveform $s_j(t)$ is transmitted in a given signaling interval after the transmission of the signal waveform $s_i(t)$ in the previous signaling interval.

The transition probabilities may be arranged in matrix form as

$$\mathbf{P} = \begin{bmatrix} p_{11} & p_{12} & \cdots & p_{1K} \\ p_{21} & p_{22} & \cdots & p_{2K} \\ \vdots & & & \\ p_{K1} & p_{K2} & \cdots & p_{KK} \end{bmatrix} \tag{8.5.7}$$

where $\mathbf{P}$ is the **transition probability matrix**.

For the NRZI signal with equal state probabilities $p_1 = p_2 = \frac{1}{2}$ and state diagram shown in Figure 8.23, the transition probability matrix is given by

$$\mathbf{P} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \end{bmatrix}. \tag{8.5.8}$$

Similarly, the transition probability matrix for the Miller-coded NRZI modulated signal with equally likely symbols $(p_1 = p_2 = p_3 = p_4 = \frac{1}{4})$ is given by

$$\mathbf{P} = \begin{bmatrix} 0 & \frac{1}{2} & 0 & \frac{1}{2} \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 \end{bmatrix}. \tag{8.5.9}$$

Modulation Code for Compact Disc System

In the design of a modulation code for the compact disc system, several factors had to be considered.

One constraint is that the maximum runlength of zeros must be sufficiently small to allow the system to synchronize from the readback signal.

To satisfy this constraint, $\kappa = 10$ was chosen.

A second factor is the frequency content of the modulated signal below 20 kHz.

To avoid interference with these control signals, the modulated information signal must fall above this frequency range.

The runlength-limited code that was implemented for this purpose is a $(d, \kappa) = (2, 10)$ code that results in a coded that encodes eight information bits into 14 coded bits and is called EFM (eight-to-fourteen modulation).

Since each audio signal sample is quantized to 16 bits, a 16-bit sample is divided into two 8-bit bytes and encoded.

By enumerating all the 14-bit codewords, one can show that there are 267 distinct codewords that satisfy the (d, κ) constraints.

Of these, 256 codewords are selected to form the code.

However, the (d, κ) constraints are not satisfied when the codewords are concatenated (merged in a sequence).

To remedy this problem, three additional bits are added to the 14, called “merging” bits.

The three merging bits serve two purposes.

First of all, if the d-constraint is not satisfied in concatenation, we choose 0's for the merging bits.

On the other hand, if the κ-constraint is being violated due to concatenation, we select one of the bits as 1.

Since two bits are sufficient to accomplish these goals, the third merging bit may be viewed as an added degree of freedom.

This added degree of freedom allows us to use the merging bits to minimize the low-frequency content of the modulated signal.

Since the merging bits carry no audio signal information, they are discarded in the readback process prior to decoding.

A measure of the low-frequency content of a digital modulated signal is the **running digital sum** (RDS), which is the difference between the total zeros and the total ones in the coded sequence accumulated from the beginning of the disc.

In addition to satisfying the (d, κ) constraints, the three merging bits are instrumental in reducing the low-frequency content below 20 kHz by an additional factor of 10 (in power).

In addition to the coded information bits and merging bits, additional bits are added for control and display (C & D), synchronization bits, and parity bits.

The data bits, control bits, parity bits, and synchronization bits are arranged in a frame srtructure, consisting of 588 bits/frame.

The frame structure is shown in Figure 8.29.

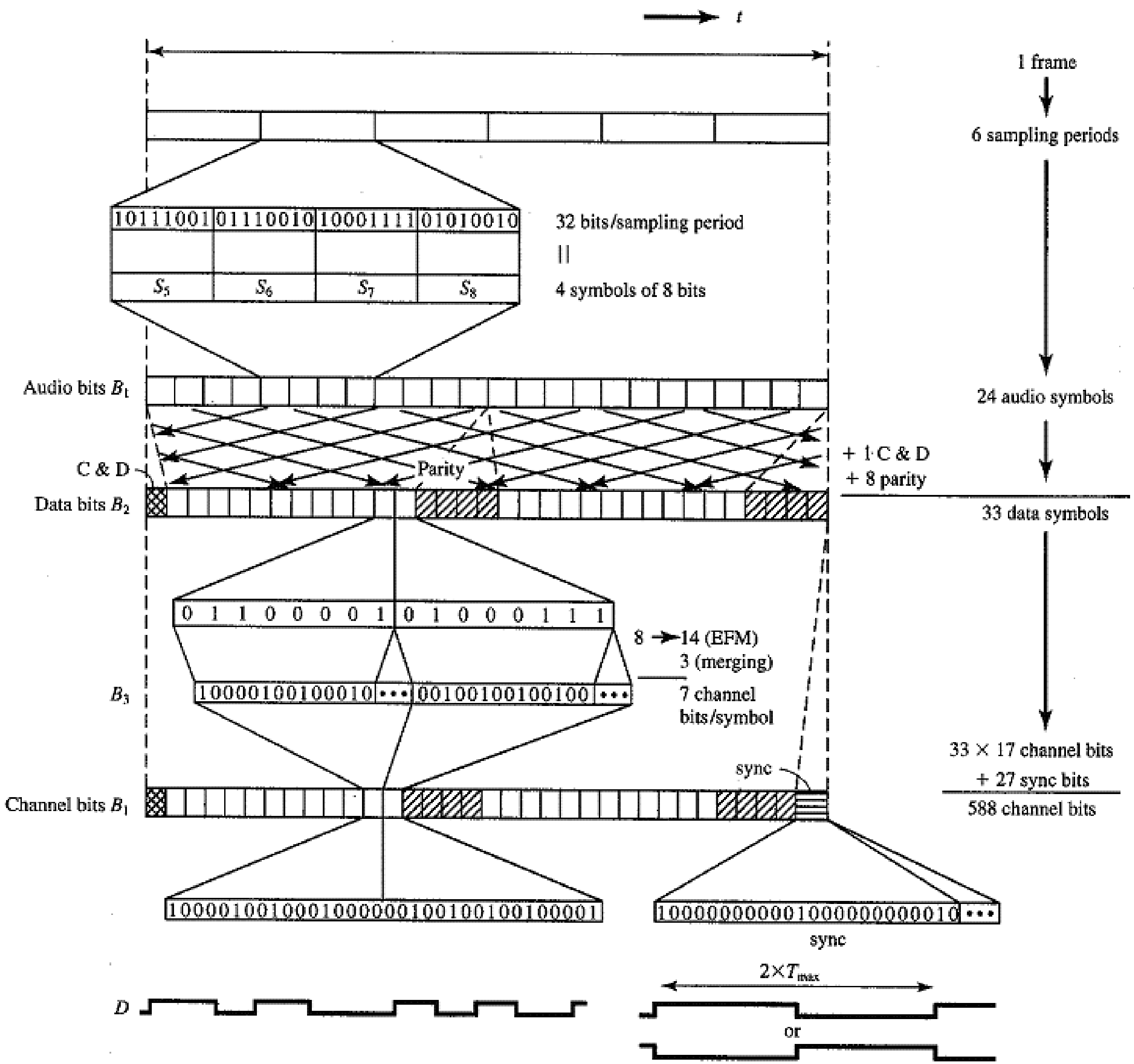


Figure 8.29 Frame structure if data in a compact disc.

8.5.2 The maximum-Likelihood Sequence Detector

When the transmitted signal has memory; i.e., the signals transmitted in successive symbol intervals are interdependent, the optimum detector bases its decisions on observation of a sequence of received signals over successive signal intervals.

Consider as an example the NRZI signal described in Section 8.5.1.
Its memory is characterized by the trellis shown in Figure 8.24.

The signal transmitted in each signal interval is binary PAM.

Hence, there are two possible transmitted signals corresponding to the signal points $s_1 = -s_2 = \sqrt{\mathcal{E}_b}$, where $\mathcal{E}_b$ is the energy/bit.

As shown in Section 7.5, the output of the matched-filter or correlation-type demodulator for binary PAM in the kth signal interval is given by

$$r_k = \pm\sqrt{\varepsilon_b} + n_k \tag{8.5.10}$$

where n_k is a zero-mean Gaussian random variable with variance $\sigma_n^2 = \dfrac{N_0}{2}$.

Consequently, the conditional PDFs for the two possible transmitted signals are given by

$$f(r_k \mid s_1) = \frac{1}{\sqrt{2\pi}\sigma_n} e^{-\frac{(r_k - \sqrt{\varepsilon_b})^2}{2\sigma_n^2}}$$

$$f(r_k \mid s_2) = \frac{1}{\sqrt{2\pi}\sigma_n} e^{-\frac{(r_k + \sqrt{\varepsilon_b})^2}{2\sigma_n^2}}. \tag{8.5.11}$$

Now, suppose we observe the sequence of matched-filter outputs $r_1, r_2, \cdots, r_K$.

Since the channel noise is assumed to be white and Gaussian, and $\psi(t-iT)$, $\psi(t-jT)$ for $i \neq j$ are orthogonal, it follows that $E(n_k n_j) = 0$, $k \neq j$.

Hence, the noise sequence $n_1, n_2, \cdots, n_k$ is also white.

Consequently, for any given transmitted sequence $\mathbf{s}^{(m)}$, the joint PDF of $r_1, r_2, \cdots, r_K$ is expressed as a product of k marginal PDFs; i.e.,

$$
\begin{aligned}
p(r_1, r_2, \cdots, r_K \,|\, \mathbf{s}^{(\mathbf{m})}) &= \prod_{k=1}^{K} p(r_k \,|\, s_k{}^{(m)}) \\
&= \prod_{k=1}^{K} \frac{1}{\sqrt{2\pi}\sigma_n} e^{\frac{(r_k - s_k{}^{(m)})^2}{2\sigma_n^2}} \\
&= \left(\frac{1}{\sqrt{2\pi}\sigma_n} \right)^K \exp\left[-\sum_{k=1}^{K} \frac{(r_k - s_k{}^{(m)})^2}{2\sigma_n^2} \right] \qquad (8.5.12)
\end{aligned}
$$

where either $s_k = \sqrt{\varepsilon_b}$ or $s_k = -\sqrt{\varepsilon_b}$.

Then, given the received sequence $r_1, r_2, \cdots, r_K$ at the output of the matched-filter or correlation-type demodulator, the detector determines the sequence $\mathbf{s}^{(\mathbf{m})} = \{s_1^{(m)}, s_2^{(m)}, \cdots, s_K^{(m)}\}$ that maximizes the conditional PDF $p(r_1, r_2, \cdots r_K \,|\, \mathbf{s}^{(m)})$.

Such a detector is called the **maximum-likelihood** (ML) **sequence detector**.

Taking the logarithm of (8.5.12) and neglecting the terms that are independent of $(r_1, r_2, \cdots r_K)$, we find that an equivalent ML sequence detector select the sequence $\mathbf{s}^{(m)}$ that minimizes the **Euclidean distance metric**:

$$D(\mathbf{r}, \mathbf{s}^{(m)}) = \sum_{k=1}^{K} (r_k - s_k^{(m)})^2 . \tag{8.5.13}$$

In searching through the trellis for the sequence that minimizes the Euclidean distance $D(\mathbf{r}, \mathbf{s}^{(m)})$, it may appear that we must compute the distance $D(\mathbf{r}, \mathbf{s}^{(m)})$ for every possible path (sequence).

For the NRZI example given above, which employs binary modulation, the total number of paths is 2^K, where K is the number of outputs obtained from the demodulator.

We may reduce the number of sequences in the trellis search by using the **Viterbi algorithm** to eliminate sequences as new data is received from the demodulator.

The Viterbi algorithm is a sequential trellis search algorithm for performing ML sequence detection to be described in detail in Chapter 9 as a decoding algorithm for channel-coded systems.

We describe it below in the context of the NRZI signal.

Assume that the search process begins initially at state S_1.

The corresponding trellis is shown in Figure 8.30.

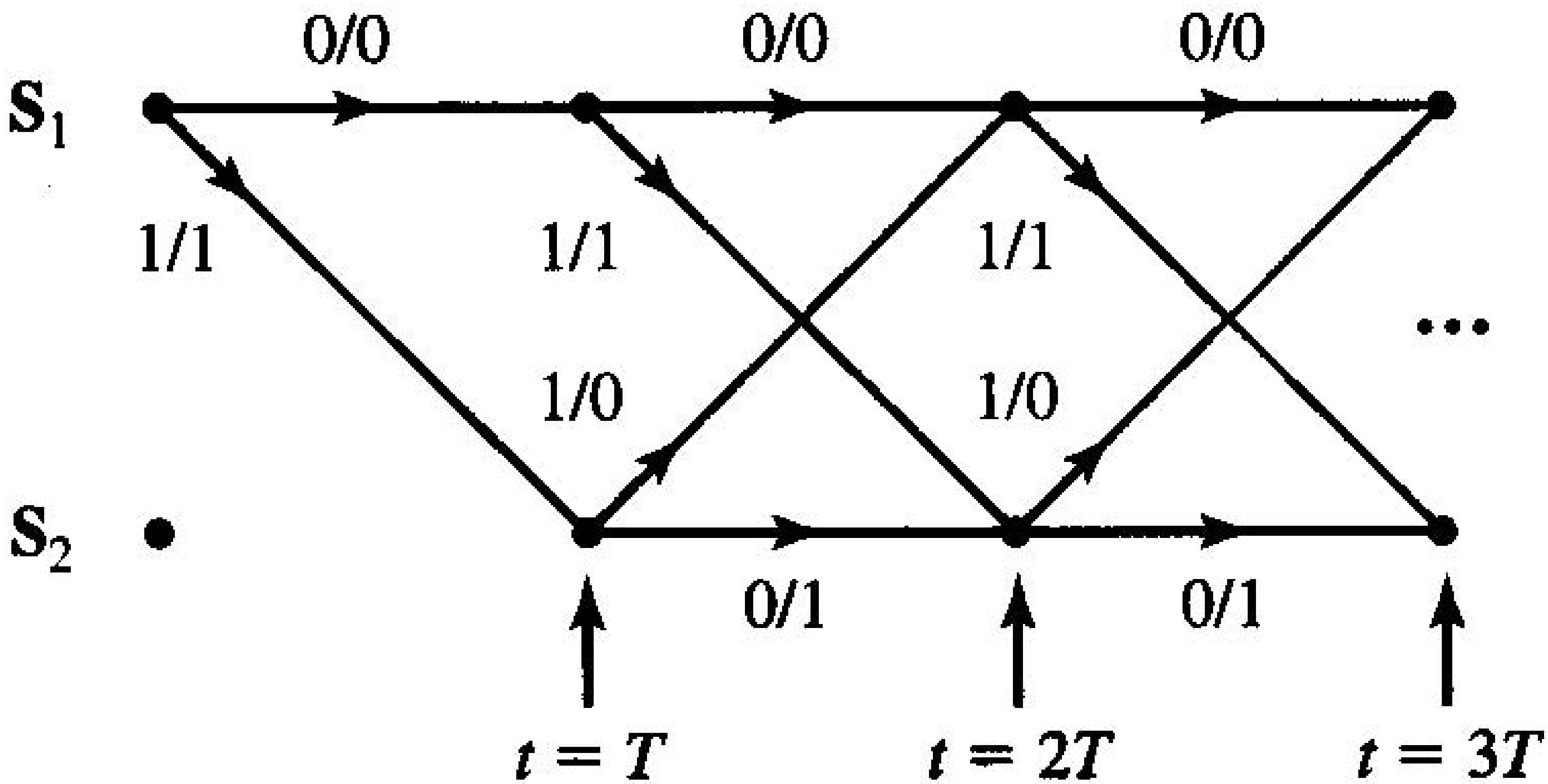


Figure 8.30 Trellis for NRZI signal with initial state S_1.

At time $t = T$, we receive $r_1 = s_1^{(m)} + n$, from the demodulator and at $t = 2T$ we receive $r_2 = s_2^{(m)} + n_2$.

Since the signal memory is one bit, which we denote as $L=1$, we observe that the trellis reaches its regular (steady-state) form after the first transition.

Thus, upon receipt of r_2 at $t=2T$ (and thereafter), we observe that there are two signal paths entering each of the two nodes and two signal paths leaving each node.

The two paths entering node S_1 at $t=2T$ correspond to the information bits $(0,0)$ and $(1,1)$ or, equivalently, to the signal points $(-\sqrt{\varepsilon_b}, -\sqrt{\varepsilon_b})$ and $(\sqrt{\varepsilon_b}, -\sqrt{\varepsilon_b})$, respectively.

The two paths entering node S_2 at $t=2T$ correspond to the information bits $(0,1)$ and $(1,0)$ or, equivalenty, to the signal points $(-\sqrt{\varepsilon_b}, \sqrt{\varepsilon_b})$ and $(\sqrt{\varepsilon_b}, \sqrt{\varepsilon_b})$, respectively.

For the two paths entering node S_1, we compute the two Euclidean distance metrics

$$\mu_2(0,0) = (r_1 + \sqrt{\varepsilon_b})^2 + (r_2 + \sqrt{\varepsilon_b})^2$$

$$\mu_2(1,1) = (r_1 - \sqrt{\varepsilon_b})^2 + (r_2 + \sqrt{\varepsilon_b})^2 \qquad (8.5.14)$$

by using the outputs r_1 and r_2 from the demodulator.

The Viterbi algorithm compares these two metrics and discards the path having the larger (greater distanc) metric.

The other path with the lower metric is saved and is called the **survivor** at $t = 2T$.

The elimination of one of the two paths may be done without compromising the optimality of the trellis search, because any extension of the path with the larger distance beyond $t = 2T$ will always have a larger metric than the survivor that is extended along the same path beyond $t = 2T$.

Similarly, for the two paths entering node S_2 at $t = 2T$, we compute the two Euclidean distance metrics

$$\mu_2(0,1) = (r_1 + \sqrt{\varepsilon_b})^2 + (r_2 - \sqrt{\varepsilon_b})^2$$

$$\mu_2(1,0) = (r_1 - \sqrt{\varepsilon_b})^2 + (r_2 - \sqrt{\varepsilon_b})^2 \tag{8.5.15}$$

by using the outputs r_1 and r_2 from the demodulator.

The two metrics are compared and the signal path with the larger metric is eliminated.

Thus, at $t = 2T$, we are left with two survivor paths, one at node S_1 and the other at node S_2, and their corresponding metrics.

The signal paths at nodes S_1 and S_2 are then extended along the two survivor paths.

Upon receipt of r_3 at $t = 3T$, we compute the metrics of the two paths entering state S_1. Suppose the survivors at $t = 2T$ are the paths $(0, 0)$ at S_1 and $(0,1)$ at S_2.

Then, the two metrics for the paths entering S_1 at $t = 3T$ are given by

$$\mu_3(0,0,0) = \mu_2(0,0) + (r_3 + \sqrt{\varepsilon_b})^2$$

$$\mu_3(0,1,1) = \mu_2(0,1) + (r_3 + \sqrt{\varepsilon_b})^2 \quad .$$

(8.5.16) These two metrics are compared and the path with the larger (distance) metric is eliminated.

Similarly, the metrics for the two paths entering S_2 at $t = 3T$ are given by

$$\mu_3(0,0,1) = \mu_2(0,0) + (r_3 - \sqrt{\varepsilon_b})^2$$

$$\mu_3(0,1,0) = \mu_2(0,1) + (r_3 - \sqrt{\varepsilon_b})^2 \qquad (8.5.17)$$

These two metrics are compared and the path with the larger (distance) metric is eliminated.

This process is continued as each new signal sample is received from the demodulator.

Thus, the Viterbi algorithm computes two metrics for the two signal paths entering a node at each stage of the trellis search and eliminates one of the next stage.

Therefore, the number of paths searched in the trellis is reduced by a factor of two at each stage.

It is relatively easy to generalize the trellis shown in Figure 8.26.

The memory of the signal is $L = 1$.

From the description of the Viterbi algorithm given above, it is unclear as to how decisions are made on the individual detected information symbols given the surviving sequences.

If we have advanced to some stage, say K, where $K >> L$, in the trellis and we compare the surviving sequences, we will find that, with probability approaching one, all surviving sequences will be identical in bit (or symbol) positions $K-5L$ and less.

In a practical implementation of the Viterbi algorithm, decisions on each information bit (or symbol) are forced after a delay of $5L$ bits (or symbol) and, hence, the surviving sequences are truncated to the $5L$ most recent bits (or symbol).

Thus, a variable delay in bit or symbol detection is avoided. The loss in performance resulting from the suboptimum detection procedure is negligible if the delay is at least $5L$.

Ex. 8.5.8

Describe the decision rule for detecting the data sequence in an NRZI signal with a Viterbi algorithm having a delay of $5L$ bits.

Solution

The trellis for the NRZI signal is shown in Figure 8.30.

In this case $L = 1$, hence the delay in bit detection is set to 5 bits.

Hence, at $t = 6T$, we will have two surviving sequences, one for each of the two states, and the corresponding metrics $\mu_6(b_1, b_2, b_3, b_4, b_5, b_6)$ and $\mu_6(b_1^{'}, b_2^{'}, b_3^{'}, b_4^{'}, b_5^{'}, b_6^{'})$.

At this stage, with probability nearly equal to one the bit b_1 will be the same as $b_1^{'}$; i.e., both surviving sequences will have a common first branch.

If $b_1 \neq b_1^{'}$, we may select the bit (b_1 or $b_1^{'}$) corresponding to the smaller of the two metrics.

Then the first bit is dropped from the two surviving sequences.

At $t = 7T$, the two metrics $\mu_7(b_2, b_3, b_4, b_5, b_{6,}\ b_7)$ and $\mu_7(b_2^{'}, b_3^{'}, b_4^{'}, b_5^{'}, b_6^{'}, b_7^{'})$ will be used to determine the decision on bit b_2.

This process continues at each stage of the search through the trellis for the minimum distance sequence.

Thus the detection delay is fixed at 5 bits.

8.5.3 Maximum-Likelihood Sequence Detection of Partial Response Signals

Partial response waveforms are signal waveforms with memory.

This memory is conveniently represented by a trellis.

For example, the trellis for the duobinary partial response signal for binary data transmission is shown in Figure 8.31.

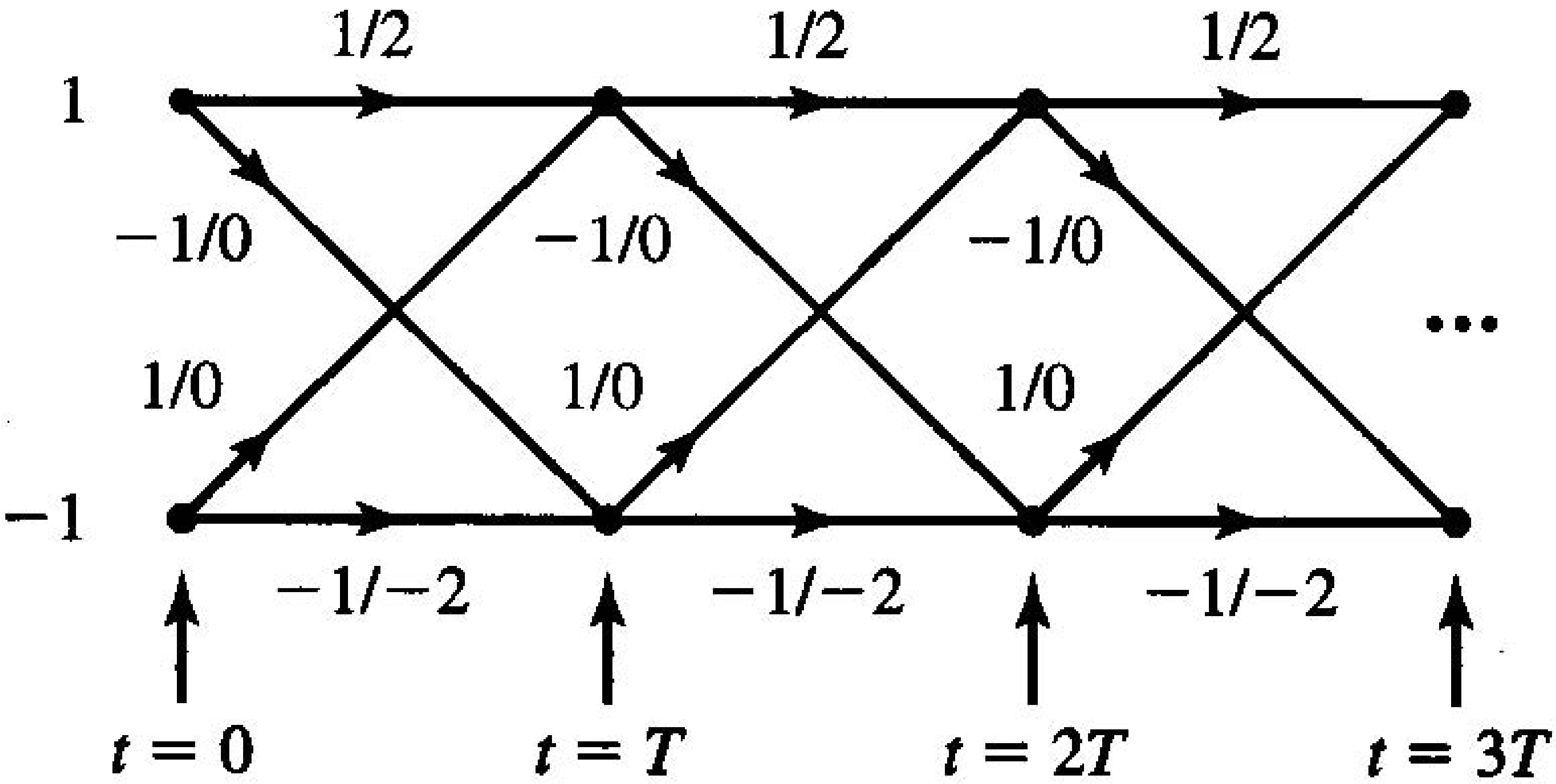


Figure 8.31 Trellis for duobinary partial response signal.

For binary modulation, this trellis contains two states, corresponding to the two possible input values of a_m; i.e., $a_m = \pm 1$.

Each branch in the trellis is labled by two numbers.

The first number on the left is the new data bit; i.e., $a_{m+1} = \pm 1$.

This number determines the transition to the new state.

The number on the right is the received signal level.

The duobinary signal has a memory of length $L = 1$.

Hence, for binary modulation the trellis has $S_t = 2^L$ states.

In general, for M -ary modulation, the number of trellis states is M^L.

The optimum ML sequence detector selects the most probable path through the trellis upon observing the received data sequence $\{y_m\}$ at the sampling instants $t = mT$, $m = 1, 2, \cdots$.

In general, each node in the trellis will have M incoming paths and M corresponding metrics.

One out of the M incoming paths is selected as the most probable, based on the values of the metrics and the other $M-1$ paths and their metrics, are discarded.

The surviving path at each node is then extended to M new paths, one for each of the M possible input symbols, and the search process continues.

This is basically the Viterbi algorithm for performing the trellis search.

For the class of partial response signals, the received sequence $\{y_m, 1 \le m \le N\}$ is generally described statistically by the joint PDF $f(\mathbf{y}_N \mid \mathbf{a}_N)$,

where $\mathbf{y}_N = (y_1, y_2, \cdots, y_N)^t$, $\mathbf{a}_N = (a_1, a_{2,} \cdots, a_N)^t$ and $N > L$.

When the additive noise is zero-mean Gaussian, $f(\mathbf{y}_N \mid \mathbf{a}_N)$ is a multivariate Gaussian PDF; i.e.,

$$f(\mathbf{y}_N \mid \mathbf{a}_N) = \frac{1}{(2\pi^{\frac{N}{2}}) \mid \det(\mathbf{C}) \mid^{\frac{N}{2}}} e^{-\frac{1}{2}(\mathbf{y}_N - \mathbf{b}_N)^t \mathbf{C}^{-1} (\mathbf{y}_N - \mathbf{b}_N)} \tag{8.5.18}$$

where $\mathbf{b}_N = (b_1, b_2, \cdots, b_N)^t$ is the mean of the vector $\mathbf{y}_N$ and

$\mathbf{C}$ is the $N \times N$ covariance matrix of $\mathbf{y}_N$.

Then, the maximum-likelihood sequence detector selects the sequence through the trellis that maximizes the PDF $f(\mathbf{y}_N \mid \mathbf{a}_N)$.

The computations for finding the most probable sequence through the trellis is simplified by taking the natural logarithms of $f(\mathbf{y}_N \mid \mathbf{a}_N)$.

Thus,

$$\ln f(\mathbf{y}_N \mid \mathbf{a}_N) = -\frac{N}{2}\ln(2\pi) - \frac{1}{2}\ln \mid \det(\mathbf{C}) \mid - \frac{1}{2}(\mathbf{y}_N - \mathbf{b}_N)^t \mathbf{C}^{-1} (\mathbf{y}_N - \mathbf{b}_N) \tag{8.5.19}$$

Given the received sequence $\{y_m\}$, the data sequence $\{a_m\}$ that maximizes $\ln f(\mathbf{y}_N \mid \mathbf{a}_N)$ is identical to the sequence $\{a_m\}$ that minimizes $(\mathbf{y}_N - \mathbf{b}_N)^t \mathbf{C}^{-1} (\mathbf{y}_N - \mathbf{b}_N)$; i.e.,

$$\hat{\mathbf{a}}_N = \arg_{\mathbf{a_N}} \min (\mathbf{y}_N - \mathbf{b}_N)^t \mathbf{C}^{-1} (\mathbf{y}_N - \mathbf{b}_N).$$

The search through the trellis for the minimum distance path may be performed sequentially by use of the Viterbi algorithm.

Let us consider the duobinary signal waveform with binary modulation and suppose that we begin at the initial state with $a_0 = 1$.

Then upon receiving $y_1 = a_1 + a_0 + v_1$ at time $t = T$ and $y_2 = a_2 + a_1 + v_2$ at time $t = 2T$, we have four candidate paths, corresponding to $(a_1, a_2) = (1, 1)$, $(-1, 1)$, $(1, -1)$, and $(-1, -1)$.

The first two candidate paths merge at state 1 at $t = 2T$.

For the two paths merging at state 1, we compute the metrics $\mu_2(1,1)$ and $\mu_2(-1,1)$ and select the more probable path.

A similar computation is performed at state -1 for the two sequences $(1,-1)$ and $(-1,-1)$.

Thus, one of the two sequences at each node is saved and the other is discarded.

The trellis search continues upon receipt of the signal samply y_3 at time $t = 3T$, by extending the two surviving paths from time $t = 2T$.

The metric computations are complicated by the correlation of the noise samples at the output of the matched filter for the partial response signal.

For example, in the case of the duobinary signal waveform, the correlation of the noise sequence $\{v_m\}$ is over two successive signal samples.

Hence, v_m and v_{m+k} are correlated for $k=1$ and uncorrelated for $k>1$.

In general, a partial response signal waveform with memory L will result in a correlated noise sequence at the output of the matched filter, which satisfies the condition $E[v_m v_{m+k}] = 0$ for $k > L$.

Ungweboeck (1974) described a sequential trellis search (Viterbi) algorithm for correlated noise (see Problem 8.26).

Some simplification in the metric computations results if we ignore the noise correlation by assuming that $E(v_m v_{m+k}) = 0$ for $k > 0$.

Then, by assumption the covariance matrix $\mathbf{C} = \sigma_v^2 \mathbf{I}$ where $\sigma_{\mathrm{v}}^2 = E[v_m^2]$ and $\mathbf{I}$ in the $N \times N$ identity matrix.

In this case, (8.5.20) simplifies to

$$\begin{aligned}\hat{\mathbf{a}}_N &= \arg_{\mathbf{a}_N} \min[(\mathbf{y}_N - \mathbf{b}_N)^t (\mathbf{y}_N - \mathbf{b}_N)] \\ &= \arg_{\mathbf{a}_N} \min\left[\sum_{m=1}^{N}\left(y_m - \sum_{k=0}^{L} x_k a_{m-k}\right)^2\right] \qquad (8.5.21)\end{aligned}$$

where

$$b_m = \sum_{k=0}^{L} x_k a_{m-k} \tag{8.5.22}$$

and $x_k = x(kT)$ are the sampled values of the partial response signal waveform.

In this case, the metric computations at each node of the trellis have the form

$$\mu_m(\mathbf{a}_m) = \mu_{m-1}(\mathbf{a}_{m-1}) + \left(y_m - \sum_{k=0}^{L} x_k a_{m-k} \right)^2 \tag{8.5.23}$$

where $\mu_m(\mathbf{a}_m)$ are the metrics at time $t = mT$, $\mu_{m-1}(\mathbf{a}_{m-1})$ are the metrics at time $t = (m-1)T$ and the second term on the right-hand side of (8.5.23) are the new increments to the metrics based on the new received samples y_m.

ML sequence detection introduces a variable delay in detecting each transmitted information symbol.

In practice, the variable delay in avoided by truncating the surviving sequences to N_t most recent symbols, where $N_t >> 5L$, thus achieving a fixed delay.

In case the M^L surviving sequences at time $t = mT$ disagree on the symbol a_{m-N_t}, the symbol in the most probable surviving sequence may be chosen.

The loss in performance resulting from this truncation is negligible if $N_t > 5L$.

Ex. 8.5.9

For the duobinary partial response signal, express the metric computations performed at $t = 2T$ and $t = 3T$, based on the received signal samples $y_m = b_m + v_m$ for $m = 1, 2, 3, $, where the noise correlation is ignored.

Solution

The metrics are generally given by (8.5.23).

Upon receipt of y_1 and y_2 and with $a_0 = 1$, the metrics for the two paths merging at state 1 are given by

$$\mu_2(1,1) = (y_1 - 2)^2 + (y_2 - 2)^2$$

$$\mu_2(-1,1) = y_1^2 + y_2^2.$$

If $\mu_2(1,1) < \mu_2(-1,1)$ we select the path (1, 1) as the more probable and discard $(-1,1)$.

Otherwise, we select the path $(-1,1)$ and discard the path (1, 1).

The path with the smaller metric is called the survivor and the sequence (a_1, a_2) and the corresponding metric are saved.

A similar computation is performed at state -1 for the sequences y_2 and $(-1,-1)$.

Then, we have

$$\mu_2(1,-1) = (y_1 - 2)^2 + y_2^2$$

$$\mu_2(-1,-1) = y_1^2 + (y_2 + 2)^2 .$$

We compare the metrics $\mu_2(1,-1)$ and $\mu_2(-1,-1)$ and select the sequence with the smaller metric.

Upon receipt of y_3 at $t = 3T$, we consider the extensions of the survivor paths.

Suppose that the two survivor paths are $(a_1, a_2) = (1,1)$ and $(a_1, a_2) = (1,-1)$.

Then, at state 1 (at $t = 3T$) we have the two merging paths $(a_1, a_2, a_3) = (1,1,1)$ and $(1,-1,1)$.

Their corresponding metrics are given by

$$\mu_3(1,1,1) = \mu_2(1,1) + (y_3 - 2)^2$$

$$\mu_3(1,-1,1) = \mu_2(1,-1) + y_3^{\,2}.$$

We compare the metrics for these two merging paths and select the path with the smaller metrics as the survivor.

Similarly, at state -1 (at $t = 3T$), we have the two merging paths $(1,1,-1)$ and $(1,-1,-1)$, and their corresponding metrics

$$\mu_3(1,1,-1) = \mu_2(1,1) + y_3^2$$

$$\mu_2(1,-1,-1) = \mu_2(1,-1) + (y_3 + 2)^2.$$

We select the path with the smaller metrics as the survivor at state -1.

This process continues upon receipt of additional data at $t = kT$, $k = 4, 5, \cdots$.

Error Probability of the Maximum-Likelihood Sequence Detector

In general, the computation of the exact probability of error is extremely difficult.

Instead, we shall determine an approximation to the probability of error, which is based on comparing the metrics of two paths which merge at a node and which are separated by the smallest Euclidean distance of all other paths.

Let us consider the trellis for the duobinary partial response signal shown in Figure 8.31.

Assume that we start in state 1 at $t = 0$ and that the first two transmitted symbols are $a_1 = 1$ and $a_2 = 1$.

Then, at $t = T$ we receive $y_1 = 2d + v_1$ and

at $t = 2T$ we receive $y_2 = 2d + v_2$.

An error is made at state 1 if the path $(a_1, a_2) = (-1, 1)$ is more probable than the path $(a_1, a_2) = (1, 1)$, given the received values of y_1 and y_2.

This path error event is the dominant path error event and, hence, it serves as a good approximation to the probability of error for the ML sequence detector.

We recall that the metric for the path $(a_1, a_2) = (1, 1)$ is given by

$$\mu_2(1,1) = [y_1 - 2d \quad y_2 - 2d]\mathbf{C}^{-1}\begin{bmatrix} y_1 - 2d \\ y_2 - 2d \end{bmatrix} \tag{8.5.24}$$

where the covariance matrix $\mathbf{C}$ is given by (see Problem 8.30)

$$\mathbf{C} = \frac{2N_0}{\pi}\begin{bmatrix} 1 & \frac{1}{3} \\ \frac{1}{3} & 1 \end{bmatrix}. \tag{8.5.25}$$

For the path $(a_1, a_2) = (-1,\ 1)$, the corresponding metric is given by

$$\mu_2(-1,\ 1) = [y_1 \quad y_2]\,\mathbf{C}^{-1}\begin{bmatrix} y_1 \\ y_2 \end{bmatrix}. \tag{8.5.26}$$

The probability of a path error event is simply the probability that the metric $\mu_2(-1,1)$ is smaller than the metric $\mu_2(1,1)$; i.e.,

$$P_2 = P[\mu_2(-1,1) < \mu_2(1,1)]. \tag{8.5.27}$$

By substituting $y_1 = 2d + v_1$ and $y_2 = 2d + v_2$ into (8.5.24) and (8.5.26) we find that

$$P_2 = P(v_1 + v_2 < -2d). \tag{8.5.28}$$

Since v_1 and v_2 are zero-mean (correlated) Gaussian variables, their sum is also zero-mean Gaussian.

The variance of the sum $z = v_1 + v_2$ is simply $\sigma_z^2 = \dfrac{16N_0}{3\pi}$.

Therefore,

$$
\begin{aligned}
P_2 &= P(z < -2d) \\
&= Q\left(\frac{2d}{\sigma_z}\right) \\
&= Q\left(\sqrt{\frac{4d^2}{\sigma_z^2}}\right).
\end{aligned}
\tag{8.5.29}
$$

From (8.4.32) we have (with $M = 2$) the expression for d^2 as

$$
\begin{aligned}
d^2 &= \frac{\pi P_{av} T}{4} \\
&= \frac{\pi \mathcal{E}_b}{4}.
\end{aligned}
\tag{8.5.30}
$$

Hence, the probability of the path error event is given by

$$
P_2 = Q\left(\sqrt{\frac{1.5\pi^2}{16}\left(\frac{2\mathcal{E}_b}{N_0}\right)}\right).
\tag{8.5.31}
$$

First, we note that this path error event results in one bit-error in the sequence of two bits.

Hence, the bit-error probability is $\frac{P_2}{2}$.

Second, there is a reduction in SNR of $10\log\frac{1.5\pi^2}{16} = -0.34\,\text{dB}$ relative to the case of no intersymbol interference.

This small SNR degradation is apparently the penalty incurred in exchange for the bandwidth efficiency of the partial response signal.

Finally, we observe that the ML sequence detector has gained back 1.76 dB of the 2.1 dB degradation inherent in the symbol-by-symbol detector.

8.5.4 Power Spectral Density of Digital Signals with Memory

In Section 8.5.1, we demonstrated that state-dependent modulation codes resulted in modulated signals with memory.

Such signals were described by Markov chains which are basically graphs that include the possible “states” of the modulator with corresponding state probabilities $\{p_i\}$, and state transitions with corresponding state transition probabilities $\{p_{ij}\}$.

The power-spectral density of digitally modulated signals that are characterized by Markov chins may be derived by following the basic procedure given in Section 8.2.

Thus, we may determine the autocorrelation function and then evaluate its Fourier transform to obtain the power-spectral density.

For signals that are generated by a Markov chain with transition probability matrix $\mathbf{P}$, as generally given by (8.5.7), the power-spectral density of the modulated signal may be expressed in the general form

$$S(f) = \frac{1}{T^2}\sum_{n=-\infty}^{\infty} \left|\sum_{i=1}^{K} p_i S_i\left(\frac{n}{T}\right)\right|^2 \delta\left(f-\frac{n}{T}\right) + \frac{1}{T}\sum_{i=1}^{K} p_i \left|S_i^{'}(f)\right|^2 + \frac{2}{T}\mathrm{Re}\left[\sum_{i=1}^{K}\sum_{j=1}^{K} p_i S_i^{'*}(f) S_j^{'}(f) P_{ij}(f)\right] \tag{8.5.32}$$

where K is the number of states of the modulator,

$S_i(f)$ is the Fourier transform of the signal waveform $s_i(t)$,

$S_i^{'}(f)$ is the Fourier transform of $s_i^{'}(t)$,

where $s_i^{'}(t) = s_i(t) - \sum_{k=1}^{K} p_k s_k(t)$, and

$P_{ij}(f)$ is the Fourier transform of the discrete-time sequence $P_{ij}(n)$, defined as

$$P_{ij}(f) = \sum_{n=1}^{\infty} p_{ij}(n) e^{-j2\pi nfT}. \tag{8.5.33}$$

The term $P_{ij}(n)$ denotes the probability that a signal $s_j(t)$ is transmitted n signaling intervals after the transmission of $s_i(t)$.

Hence, $\{p_{ij}(n)\}$ are the transition probabilities in the transition probability matrix $\mathbf{P}^n$.

Note that when $n=1$, $\mathbf{P}^n \equiv \mathbf{P}$, which is the matrix given by (8.5.7).

Ex. 8.5.10

Determine the power-spectral density of the NRZI signal.

Solution

The NRZI signal is characterized by the transition probability matrix

$$\mathbf{P} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \end{bmatrix}.$$

We note that $\mathbf{P}^n = \mathbf{P}$ for all $n > 1$.

Hence, with $K = 2$ states and $g_T(t) = s_1(t) = -s_2(t)$, we obtain

$$S(f) = \frac{(2p-1)^2}{T^2} \sum_{n=-\infty}^{\infty} \left| G_T\left(\frac{n}{T}\right) \right|^2 \delta\left(f - \frac{n}{T} \right) + \frac{4p(1-p)}{T} |G_T(f)|^2 \tag{8.5.34}$$

where for a rectangular pulse of amplitude A,

$$|G_T(f)|^2 = (AT)^2 \left(\frac{\sin \pi fT}{\pi fT} \right)^2$$

$$= (AT)^2 \operatorname{sinc}^2(fT).$$

We observe that when $p = \frac{1}{2}$ (equally probable signals) the impulse spectrum vanishes and $S(f)$ reduces to

$$S(f) = \frac{1}{T} |G_T(f)|^2.$$

We observe that the power spectral density of the NRZI signal for equally probable signals is identical to the expression in (8.2.18), which applies to an uncorrelated sequence $\{a_n\}$ into the modulator.

Hence, we conclude that the simple precoding operation in NRZI does not result in a correlated sequence.

Ex. 8.5.11

Determine the power spectral density of the delay modulated (Miller encoded) signal described in Section 8.5.1.

Solution

The transition probability matrix of a delay modulated signal is [see (8.5.9)]

$$\mathbf{P} = \begin{bmatrix} 0 & \frac{1}{2} & 0 & \frac{1}{2} \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 \end{bmatrix}.$$

The state probabilities are $p_i = \frac{1}{4}$, for $i = 1, 2, 3, 4$.

Powers of $\mathbf{P}$ are easily obtained by use of the relation (see Problem 8.8)

$$\mathbf{P}^4 \gamma = -\frac{1}{4}\gamma \tag{8.5.35}$$

where γ is the signal correlation matrix with elements

$$\gamma_{ij} = \frac{1}{T}\int_0^T s_i(t) s_j(t) dt$$

and the four signals $\{s_i(t), i = 1, 2, 3, 4\}$ are shown in Figure 8.27, where $s_3(t) = -s_2(t)$ and $s_4(t) = -s_1(t)$.

It is easily seen that

$$\gamma = \begin{bmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & -1 & 0 \\ 0 & -1 & 1 & 0 \\ -1 & 0 & 0 & 1 \end{bmatrix}. \tag{8.5.36}$$

Consequently, powers of $\mathbf{P}$ can be generated from the relation

$$\mathbf{P}^{k+4}\gamma = -\frac{1}{4}\mathbf{P}^k\gamma, \qquad k \geq 1. \tag{8.5.37}$$

With the aid of these relations, the power spectral density of the delay modulated signal is obtained from (8.5.32).

It may be expressed in the form

$$S(f) = \frac{1}{2(\pi fT)^2(17 + 8\cos 8\pi fT)}[23 - 2\cos \pi fT - 22\cos 2\pi fT$$

$$-12\cos 3\pi fT + 5\cos 4\pi fT + 12\cos 5\pi fT + 2\cos 6\pi fT - 8\cos 7\pi fT + 2\cos 8\pi fT)]. \tag{8.5.38}$$

The power-spectral densities of the NRZI and the delay modulated signal are shown in Figure 8.32.

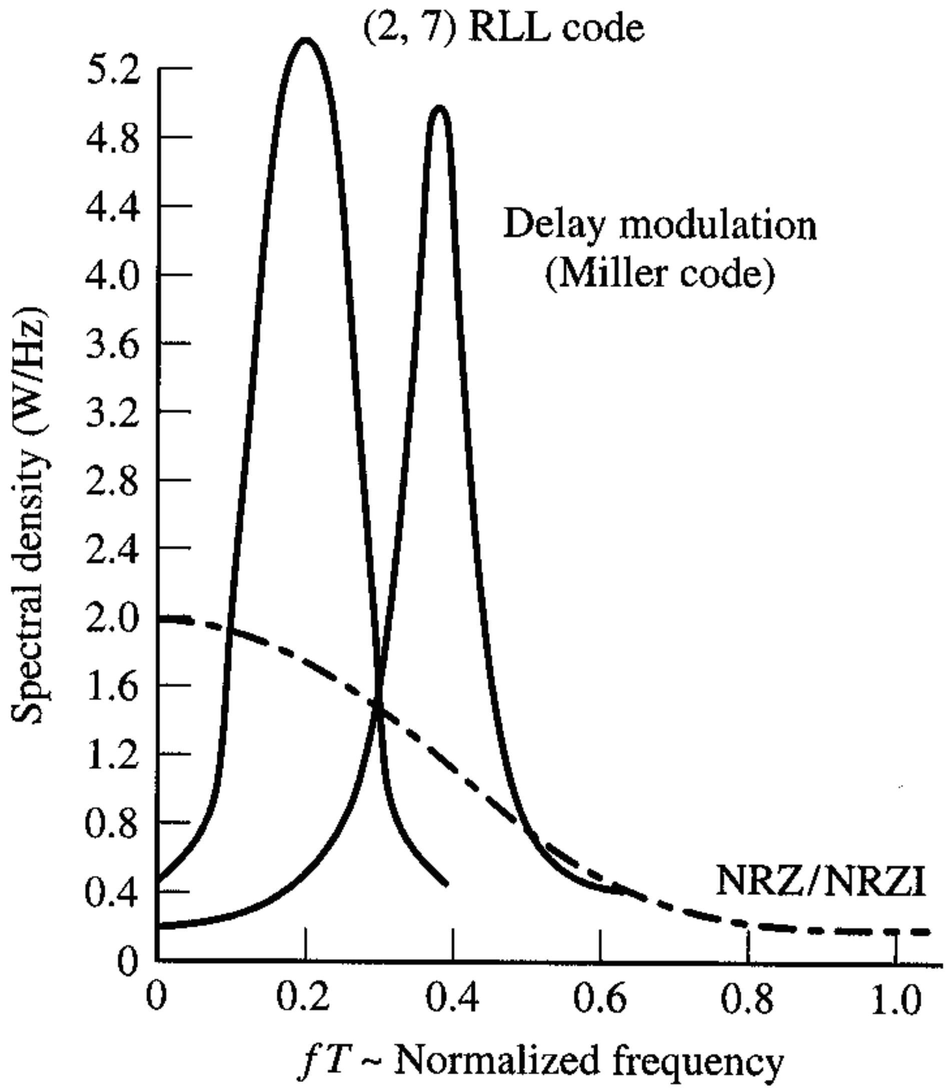


Figure 8.32 Power-spectral density (one-sided) of Miller code (delay modulation), and NRZ/NRZI baseband signals.

We observe that the NRZI signal has a lowpass power spectral density with a peak of $f = 0$.

On the other hand, the delay modulated signal has very little power in the vicinity of $f = 0$.

It also has a relatively narrow power-spectral density.

These two characteristics make it particularly suitable for use in magnetic recording storage channels which use flux-sensing heads for writing on and reading off a disk.

Also shown for comparison in Figure 8.32 is the power-spectral density of the modulated signal generated by a $(d, \kappa) = (2, 7)$ code followed by NRZI.

The $(2, 7)$ runlength-limited code, previously given in Table 8.5, is also widely used in magnetic recording channels.

When there is no memory in the modulation method, the signal waveform transmitted in each signaling interval is independent of the waveforms transmitted in previous signaling intervals.

The power-spectral density of the resulting signal may still be expressed in the form of (8.5.32) if the transition probability matrix is replaced by

$$\mathbf{P} = \begin{bmatrix} p_1 & p_2 & \cdots & p_K \\ p_1 & p_2 & \cdots & p_K \\ \vdots & \vdots & & \vdots \\ p_1 & p_2 & \cdots & p_K \end{bmatrix} \tag{8.5.39}$$

and we impose the condition that $\mathbf{P}^n = \mathbf{P}$ for all $n > 1$, as a consequence of the memoryless modulation.

Under these conditions, the expression for $S(f)$ given (8.5.32) reduces to the simple form

$$S(f) = \frac{1}{T^2} \sum_{n=-\infty}^{\infty} \left| \sum_{i=1}^{K} p_i S_i\left(\frac{n}{T}\right) \right|^2 \delta\left(f - \frac{n}{T}\right) + \frac{1}{T} \sum_{i=1}^{K} p_i (1 - p_i) |S_i(f)|^2 - \frac{2}{T} \sum_{i=1}^{K} \sum_{\substack{j=1 \\ i<j}}^{K} p_i p_j \operatorname{Re}[S_i(f) S_j^*(f)]. \tag{8.5.40}$$

We observe that our previous result for the power-spectral density of memoryless PAM modulation given by (8.2.17) is a special case of the expression in (8.5.40).

Specifically, if we have $K = M$ signals which are amplitude-scaled versions of a basic pulse $g_T(t)$; i.e., $s_m(t) = a_m g_T(t)$, where a_m is the signal amplitude, then $S_m(f) = a_m G_T(f)$ and (8.5.40) becomes

$$S(f) = \frac{1}{T^2}\left(\sum_{i=1}^{M} a_i p_i\right)^2 \sum_{n=-\infty}^{\infty} |G_T\left(\frac{n}{T}\right)|^2 \delta\left(f - \frac{n}{T}\right) + \frac{1}{T}\left[\sum_{i=1}^{M} p_i(1-p_i)a_i^2 - 2\sum_{i=1}^{M}\sum_{j=1}^{M} p_i p_j a_i a_j\right] |G_T(f)|^2. \qquad (8.5.41)$$

If we compare (8.2.17) with (8.5.41), we find that these expressions are identical, where the mean and variance of the information sequence is given by

$$m_a = \sum_{i=1}^{M} p_i(1-p_i)a_i^2 - 2\sum_{i=1}^{M}\sum_{\substack{j=1 \\ i<j}}^{M} p_i p_j a_i a_j .$$

Therefore, (8.5.40) is the more general form for the power-spectral density of memoryless modulated signals, since it applies to signals that may have different pulse shapes.

8.6 System Design in the Presence of Channel Distortion

Recall that a signal pulse $x(t)$ will satisfy the condition of zero ISI at the sampling instants $t = nT$, $n = \pm 1, \pm 2, \cdots,$ if its spectrum $X(f)$ satisfies the condition given by (8.3.9).

From this condition we concluded that for ISI free transmission over a channel, the transmitter-receiver filters and the channel transfer function must satisfy

$$G_T(f)C(f)G_R(f) = X_{rc}(f) \tag{8.6.1}$$

where $X_{rc}(f)$ denotes the Fourier transform of an appropriate raised cosine pulse whose parameters depend on the channel bandwidth W and the transmission interval T.

Obviously, there are an infinite number of transmitter-receiver filter pairs that satisfy the above condition.

In this section, we are concerned with the design of a digital communication system that suppresses ISI in a channel with distortion.

We first present a brief coverage of various types of channel distortion and then we consider the design of transmitter and receiver filters.

We distinguish two types of distortion.

Amplitude distortion results when the ampliyude characteristic $|C(f)|$ is not constant for $|f| \leq W$. **Phase distortion** results when the phase characteristic $\Theta_c(f)$ is nonlinear in frequency.

Another view of phase distortion is obtained by considering the derivative of $\Theta_c(f)$.

Thus, we define the **envelope delay** characteristic as (see Problem 2.57 and 8.40)

$$\tau(f) = -\frac{1}{2\pi}\frac{d\Theta_c(f)}{df}. \tag{8.6.2}$$

When $\Theta_c(f)$ is linear in f, the envelope delay is constant for all frequencies.

In this case, all frequencies in the transmitted signal pass through the channel with the same fixed-time delay.

In such a case, there is no phase distortion.

However, when $\Theta_c(f)$ is nonlinear, the envelope delay $\tau(f)$ varies with frequency and the various frequency components in the input signal undergo different delays in passing through the channel.

In such a case we say that the transmitted signal has suffered from **delay distortion**.

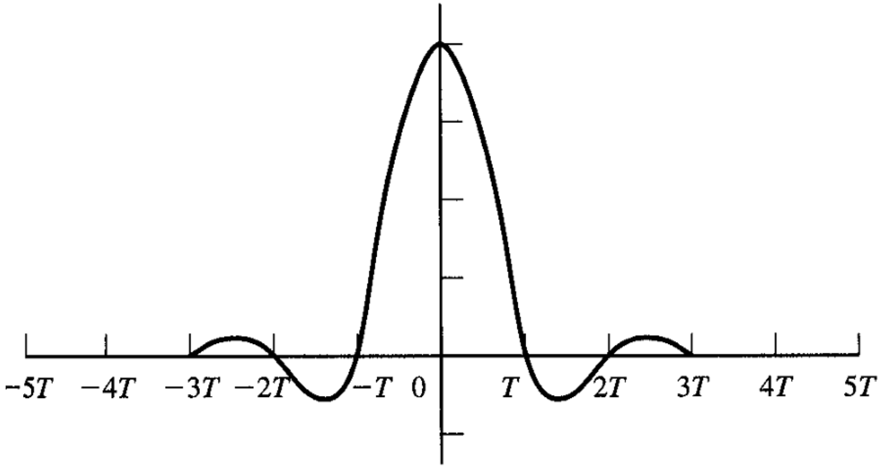
Both amplitude and delay distortion cause intersymbol interference in the received signal.

For example, let us assume that we have designed a pulse with a raised cosine spectrum that has zero ISI at the sampling instants.

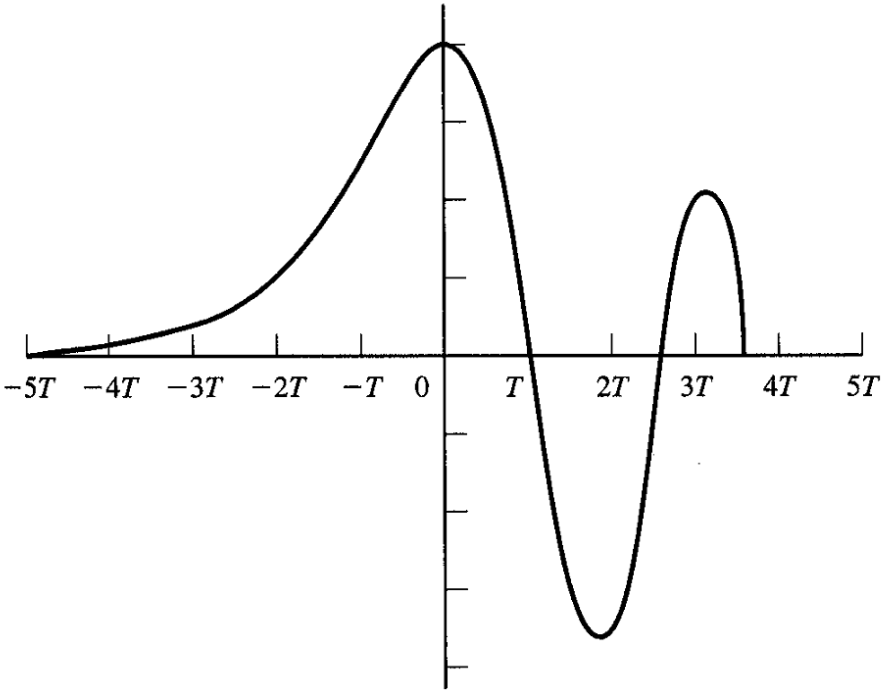
An example of such a pulse is shown in Figure 8.33(a).

When the pulse is passed through a channel filter with constant amplitude $|C(f)|=1$ for $|f|<W$ and a

quadratic phase characteristic (linear envelope delay), the received pulse at the output of the channel is shown in Figure 8.33(b).



(a)



(b)

Figure 8.33 Effect of channel distortion in (a) channel input and (b) channel output.

Note that the periodic zero crossing have been shifted by the delay distortion, so that the resulting pulse suffers from ISI.

Consequently, a sequence of successive pulse would be smeared into one another and the peaks of the pulses would no longer be distinguishable due to the ISI.

Next, we consider two problems.

First, we consider the design of transmitting and receiving filters in the presence of channel distortion when the channel characteristics are known.

Second, we consider the design of special filters, called channel equalizers, that automatically and adaptively correct for the channel distortion when the channel characteristics; i.e., $|C(f)|$ and $\angle\Theta_c(f)$, are known.

8.6.1 Design of Transmitting and Receiving Filters for a Known Channel

In this section, we assume that the channel frequency response characteristic $C(f)$ is known and consider the problem of designing a transmitting filter and a receiving filter that minimize the SNR at the output of the receiving filter and results in zero ISI.

Figure 8.34 shows the overall system under consideration.

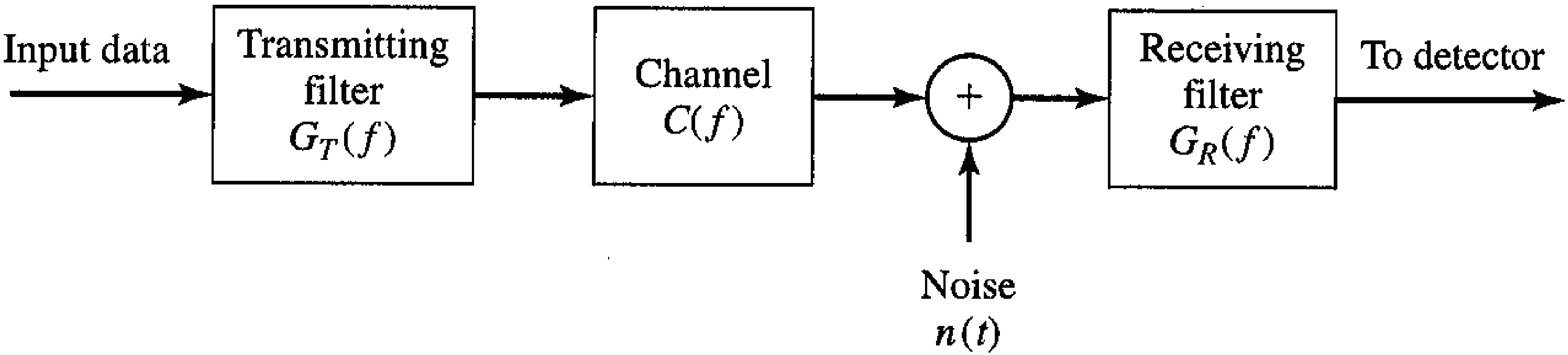


Figure 8.34 System configuration for design of $G_T(f)$ and $G_R(f)$.

For the signal component, we must satisfy the condition

$$G_T(f)C(f)G_R(f) = X_{rc}(f)e^{-j\pi f t_0}, \qquad |f| \le W, \tag{8.6.3}$$

where $X_{rc}(f)$ is the desired raised cosine spectrum that yields zsro ISI at the sampling instants, and

t_0 is a time delay which is necessary to ensure the physical realizability og the transmitter and receiver filters.

The noise at the output of the receiving filter may be expressed as

$$v(t) = \int_{-\infty}^{\infty} n(t-\tau) g_R(\tau) d\tau \tag{8.6.4}$$

where $n(t)$ is the input to the filter.

The noise $n(t)$ is assumed to be zero-mean Gaussian.

Hence, $v(t)$ is zero-mean Gaussian, with a power-spectral density

$$S_v(f) = S_n(f)|G_R(f)|^2 \tag{8.6.5}$$

where $S_n(f)$ is the spectral density of the noise process $n(t)$.

For simplicity, we consider binary PAM transmission.

Then, the sampled output of the matched filter is

$$y_m = x_0 a_m + v_m$$

$$= a_m + v_m \tag{8.6.6}$$

where x_0 is normalized to unity,

$a_m = \pm d$, and

v_m represents the noise term which is zero-mean Gaussian with variance

$$\sigma_v^2 = \int_{-\infty}^{\infty} S_n(f) \, | G_R(F) |^2 \, df \; . \tag{8.6.7}$$

Consequently, the probability of error is given by

$$P_2 = \frac{1}{\sqrt{2\pi}} \int_{\frac{d}{\sigma_v}}^{\infty} e^{-\frac{y^2}{2}} dy$$

$$= Q\left(\sqrt{\frac{d^2}{\sigma_v^2}} \right). \tag{8.6.8}$$

Now, suppose that we select the filter at the transmitter to have the frequency response

$$G_T(f) = \frac{\sqrt{X_{rc}(f)}}{C(f)} e^{-j2\pi f t_0} \tag{8.6.9}$$

where t_0 is a suitable delay to ensure causality.

Then, the cascade of the transmit filter and the channel results in the frequency response

$$G_T(f)C(f) = \sqrt{X_{rc}(f)}\, e^{-j2\pi f t_0}. \tag{8.6.10}$$

In the presence of additive white Gaussian noise, the filter at the receiver is designed to be matched to the received signal pulse.

Hence, its frequency response

$$G_R(f) = \sqrt{X_{rc}(f)}\, e^{-j2\pi f t_r} \tag{8.6.11}$$

where t_r is an appropriate delay.

Let us compute the SNR $\frac{d^2}{\sigma_v^2}$ for these filter characteristics.

The noise variance is

$$\begin{aligned}\sigma_v^2 &= \frac{N_0}{2}\int_{-\infty}^{\infty} |G_R(f)|^2\, df \\ &= \frac{N_0}{2}\int_{-W}^{W} X_{rc}(f)df \\ &= \frac{N_0}{2}\end{aligned} \tag{8.6.12}$$

The average transmitted power is given by

$$P_{av} = \frac{E(a_m^2)}{T}\int_{-\infty}^{\infty} g_T^2(t)dt$$

$$= \frac{d^2}{T}\int_{-W}^{W} \frac{X_{rc}(f)}{|C(f)|^2}df \tag{8.6.13}$$

and, hence,

$$d^2 = P_{av}T\left[\int_{-W}^{W} \frac{X_{rc}(f)}{|C(f)|^2}df\right]^{-1} \tag{8.6.14}$$

Therefore, the SNR $\dfrac{d^2}{\sigma_v^2}$ is given as

$$\frac{d^2}{\sigma_v^2} = \frac{2P_{av}T}{N_0}\left[\int_{-W}^{W} \frac{X_{rc}(f)}{|C(f)|^2} df\right]^{-1} . \tag{8.6.15}$$

We note that the term,

$$10\log_{10}\int_{-W}^{W} \frac{X_{rc}(f)}{|C(f)|^2} df \tag{8.6.16}$$

with $|C(f)| \le 1$, for $|f| \le W$, represents the loss in performance in dB of the communication system due to channel distortion.

When the channel is ideal, $|C(f)| = 1$ for $|f| \le W$ and, hence, there is no performance loss.

We also note that this loss is entirely due to amplitude distortion in the channel, because the phase distortion has been totally compensated by the transmit filter.

Ex. 8.6.1

Determine the magnitude of the transmitting and receiving filter characteristics for a binary communication system that transmits data at a rate of 4800 bits/sec over a channel with frequency (magnitude) response

$$|C(f)| = \frac{1}{\sqrt{1+\left(\frac{f}{W}\right)^2}}, \qquad |f| \le W,$$

where $W = 4800\ Hz$.

The additive noise is zero-mean, white, Gaussian with spectral density $\frac{N_0}{2} = 10^{-15}\ \mathrm{W/Hz}$.

Solution

Since $W = \frac{1}{T} = 4800$ we use a signal pulse with a raised cosine spectrum and $\alpha = 1$.

Thus,

$$X_{rc}(f) = \frac{T}{2}\left[1+\cos(\pi T\,|f|)\right]$$

$$= T\cos^2\left(\frac{\pi |f|}{9600}\right).$$

Then,

$$|G_T(f)| = \sqrt{T\left[1+\left(\frac{f}{W}\right)^2\right]}\cos\frac{\pi|f|}{9600}, \qquad |f| \le 4800\,\mathrm{Hz},$$

$$|G_R(f)| = \sqrt{T}\cos\frac{\pi|f|}{9600}, \qquad |f| \le 4800\,\mathrm{Hz},$$

and $|G_T(f)| = |G_R(f)| = 0$ for $|f| > 4800\,\mathrm{Hz}$

8.6.2 Channel Equalization

To design these transmitting and receiving filters for zero ISI at the sampling instants, the channel must be precisely known and its characteristic should not change with time.

In practice we often encounter channels whose frequency response characteristics are either unknown or change with time.

For example, in data transmission over the dial-up telephone network, the communication channel will be different every time we dial a number, because the channel route will be different.

Once a connection is made, however, the channel will be time-invariant for a relatively long period of time.

This is an example of a channel whose characteristics are unknown a priori.

Examples of time-varying channels are radio channels, such as ionospheric propagation channels.

These channels are characterized by time-varying frequency response characteristics.

These types of channels are examples where the optimization of the transmitting and receiving filters, as described in Section 8.6.1, is not possible.

Under these circumstances, we may design the transmitting filter to have a square-root raised cosine frequency response; i.e.,

$$G_T(f) = \begin{cases} \sqrt{X_{rc}(f)}e^{-j2\pi f t_0}, & |f| \leq W, \\ 0, & |f| \leq W, \end{cases}$$

and the receiving filter, with frequency response $G_R(f)$, to be matched to $G_T(f)$.

Therefore,

$$|G_T(f)||G_R(f)| = X_{rc}(f). \tag{8.6.17}$$

Then, due to channel distortion, the output of the receiving filter is given by

$$y(t) = \sum_{n=-\infty}^{\infty} a_n x(t-nT) + v(t) \tag{8.6.18}$$

where $x(t) = g_T(t) * c(t) * g_R(t)$.

The filter output may be sampled periodically to produce the sequence

$$
\begin{aligned}
y_m &= \sum_{n=-\infty}^{\infty} a_n x_{m-n} + v_m \\
&= x_0 a_m + \sum_{\substack{n=-\infty \\ n \neq m}}^{+\infty} a_n x_{m-n} + v_m
\end{aligned}
\tag{8.6.19}
$$

where $x_n = x(nT),\ n = 0, \pm 1, \pm 2, \cdots$.

The middle term on the right-hand side of (8.6.19) represents the ISI.

In any practical system, it is reasonable to assume that the ISI affects a finite number of symbols.

Hence, we may assume that $x_n = 0$ for $n < -L_1$ and $n > L_2$,

where L_1 and L_2 are finite, positive integers.

Consequently, the ISI observed at the output of the receiving filter may be viewed as being generated by passing the data sequence $\{a_m\}$ through an FIR filter with coefficients $\{x_n, -L_1 \le n \le L_2\}$, as shown in Figure 8.35 which is called the **equivalent discrete-time channel filter**.

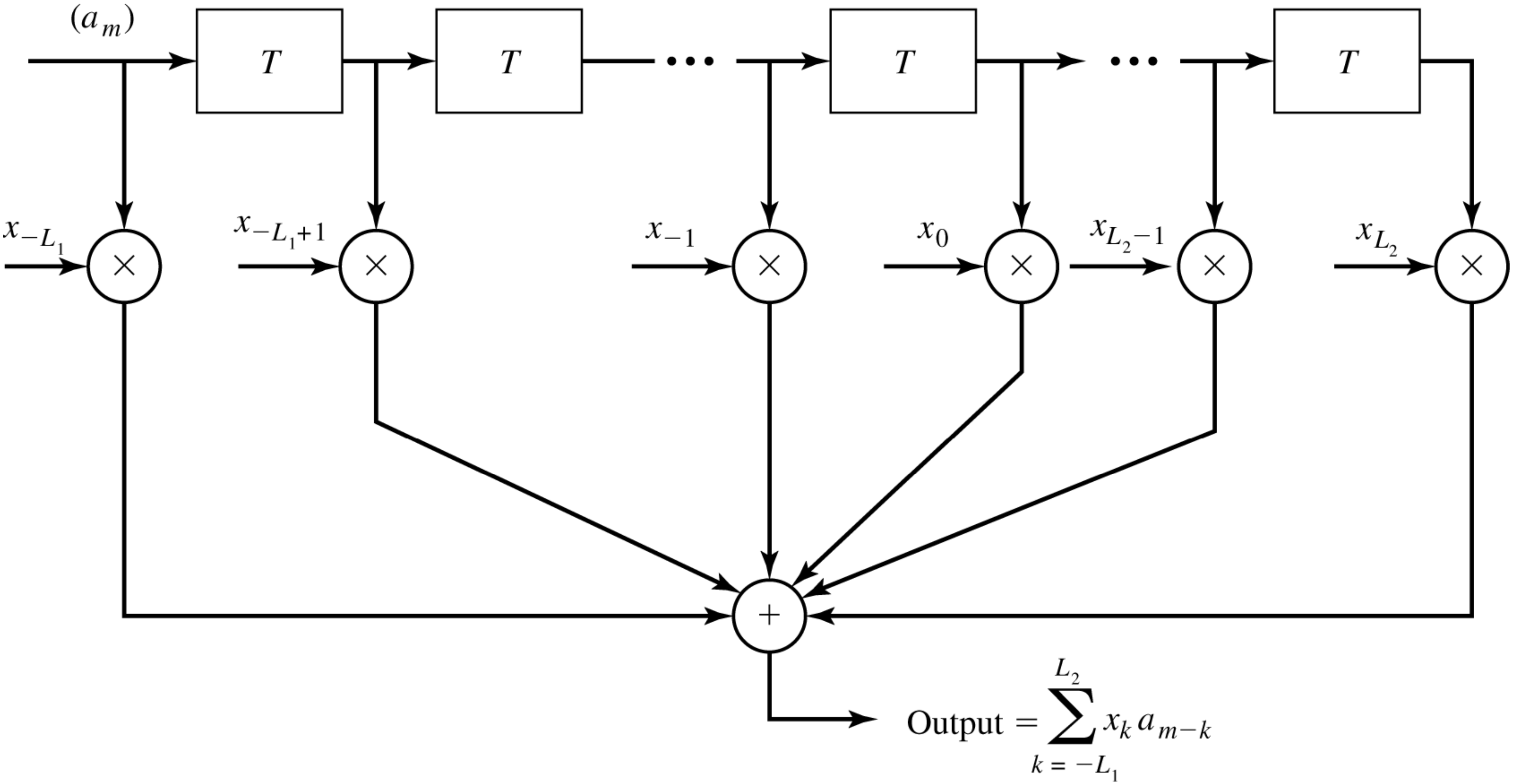


Figure 8.35

Equivalent discrete-time channel filter.

Since its input is the discrete information sequence (binary or M-art), the output of the discrete-time channel filter may be characterized as the output of a finite-state machine corrupted by additive Gaussian noise.

Hence, the noise-free output of the filter is described by a trellis having M^L states where $L = L_1 + L_2$.

Maximum-Likelihood Sequence Detection

The optimum detector for the information sequence $\{a_m\}$ based oon the observation of the received sequence $\{y_m\}$, given by (8.6.19), is a ML sequence detector.

The detector is akin to the ML sequence detector described in the context of detecting partial response signals which have controlled ISI.

The Viterbi algorithm provides a method for searching through the trllis for the ML signal path.

To accomplish this search, the equivalent channel filter coefficients $\{x_n\}$ must be known or measured by some method.

At each stage of the trellis search, there are M^L corresponding Euclidean distance path metrics.

Due to exponential increase in the computational complexity of the Viterbi algorithm with the span (length L) of the ISI, this type of detection is practical only when M and L are small.

For example in mobile cellular telephone systems which employ digital transmission of speech signals, M is usually selected to be small; e.g., $M = 2$ or 4, and $2 \le L \le 5$.

In this case, the ML sequence detector may be implemented with reasonable complexity. However, when M and L are large, the ML sequence detector becomes impractical.

In such a case other more practical but suboptimum methods are used to detect the information sequence $\{a_m\}$ in the presence of ISI.

Neverthless, the performance of the ML sequence detector for a channel with ISI serves as a benchmark for comparing its performance with that of suboptimum methods.

Two suboptimum methods are ddescribed below.

Linear Equalizers

To compensate for the channel distortion, we may employ a linear filter with adjustable parameters which is called **channel equalizers** or, simply, **equalizers**.

The filter parameters are adjusted on the basis of measurements of the channel characteristics.

On channels whose frequency-response characteristics are unknown, but time-invariant, we may measure the channel characteristics, adjust the parameters of the equalizer, and once adjusted, the parameters remain fixed during the transmission of data.

Such equalizers are called **preset equalizers**.

On the other hand, **adaptive equalizers** update their parameters on a periodic basis during the transmission of data.

First, we consider the design characteristics for a linear equalizer from a frequency domain viewpoint.

Figure 8.36 shows a block diagram of a system that employs a linear filter as a channel equalizer.

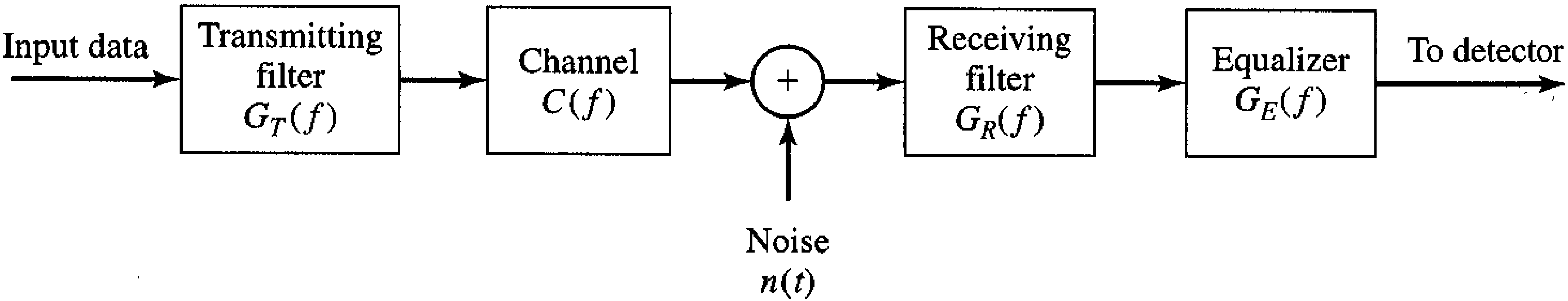


Figure 8.36 Block diagram of a system with an equalizer.

The demodulator consists of a receiving filter with frequency response $G_R(f)$ in cascade with a channel equalizing filter that has a frequency response $G_E(f)$.

Since $G_R(f)$ is matched to $G_T(f)$ and they are designed so that their product satidfies (8.6.17), $|G_E(f)|$ must compensate for the channel distortion.

Hence, the equalizer frequency response must equal the inverse of the channel response; i.e.,

$$G_E(f) = \frac{1}{C(f)} = \frac{1}{|C(f)|} e^{-j\Theta_c(f)}, \quad |f| \le W, \tag{8.6.20}$$

where $|G_E(f)| = \dfrac{1}{|C(f)|}$ and the equalizer phase characteristic $\Theta_E(f) = -\Theta_c(f)$.

In this case, the equalizer is said to be the **inverse channel filter** to the channel response.

We note that the inverse channel filter completely eliminates ISI caused by the channel. Since it forces the ISI to be zero at the sampling times $t = nT$, the equalizer is called a **zero-forcing equalizer**.

Hence, the input to the detector is of the form

$$y_m = a_m + v_m$$

where v_m is the noise component, which is zero-mean Gaussian with a variance

$$\sigma_v^2 = \int_{-\infty}^{\infty} S_n(f)\,|G_R(f)|^2 |G_E(f)|^2\, df$$

$$= \int_{-W}^{W} \frac{S_n(F)\,|X_{rc}(f)|}{|C(f)|^2} df \tag{8.6.21}$$

where $S_n(f)$ is the power-spectral density of the noise.

When the noise is white, $S_n(f) = \dfrac{N_0}{2}$ and the variance becomes

$$\sigma_v^2 = \frac{N_0}{2} \int_{-W}^{W} \frac{|X_{rc}(f)|}{|C(f)|^2} df\ . \tag{8.6.22}$$

In general, the noise variance at the output of the zero-forcing equalizer is higher than the noise variance at the output of the optimum receiving filter $|G_R(f)|$ given by (8.6.12) for the case in which the channel is known.

Ex. 8.6.2

The channel given in Example 8.6.1 is equalized by a zero-forcing equalizer.

Assuming that the transmitting and receiving filters satisfy (8.6.17), determine the value of the noise variance at the sampling instants and the probability of error.

Solution

When the noise is white, the variance of the noise at the output of the zero-forcing equalizer (input to the detector) is given by (8.6.22).

Hence,

$$\sigma_v^2 = \frac{N_0}{2}\int_{-W}^{W} \frac{|X_{rc}(f)|}{|C(f)|^2} df$$

$$= \frac{TN_0}{2}\int_{-W}^{W} \left[1+\left(\frac{f}{W}\right)^2\right] \cos^2 \frac{\pi |f|}{2W} df$$

$$= N_0 \int_0^1 (1+x^2)\cos^2 \frac{\pi x}{2} dx$$

$$= \left(\frac{2}{3}-\frac{1}{\pi^2}\right) N_0 .$$

The average transmitted power is given by

$$P_{av} = \frac{(M^2-1)d^2}{3T}\int_{-W}^{W}|G_T(f)|^2 df$$

$$= \frac{(M^2-1)d^2}{3T}\int_{-W}^{W}|X_{rc}(f)|\,df$$

$$= \frac{(M^2-1)d^2}{3T}.$$

The general expression for the probability of error is given as

$$P_M = \frac{2(M-1)}{M} Q\left(\sqrt{\frac{3P_{av}T}{(M^2-1)\left(\frac{2}{3}-\frac{1}{\pi^2}\right)N_0}}\right).$$

If the channel were ideal, the argument of the Q-function woule be $\frac{6P_{av}T}{(M^2-1)N_0}$.

Hence, the loss in performance due to the nonideal channel is given by the factor $2\left(\frac{2}{3}-\frac{1}{\pi^2}\right)=1.133$ or 0.54 dB.

Consider the design of a linear equalizer from a time-domain view-point.

We noted previously that in real channels, the ISI is limited to a finite number of samples, say L samples.

As a consequence, in practice the channel equliaer is approximated by a finite duration impulse response (FIR) filter, or transversal filter, with adjustable tap coefficients $\{c_n\}$, as shown in Figure 8.37.

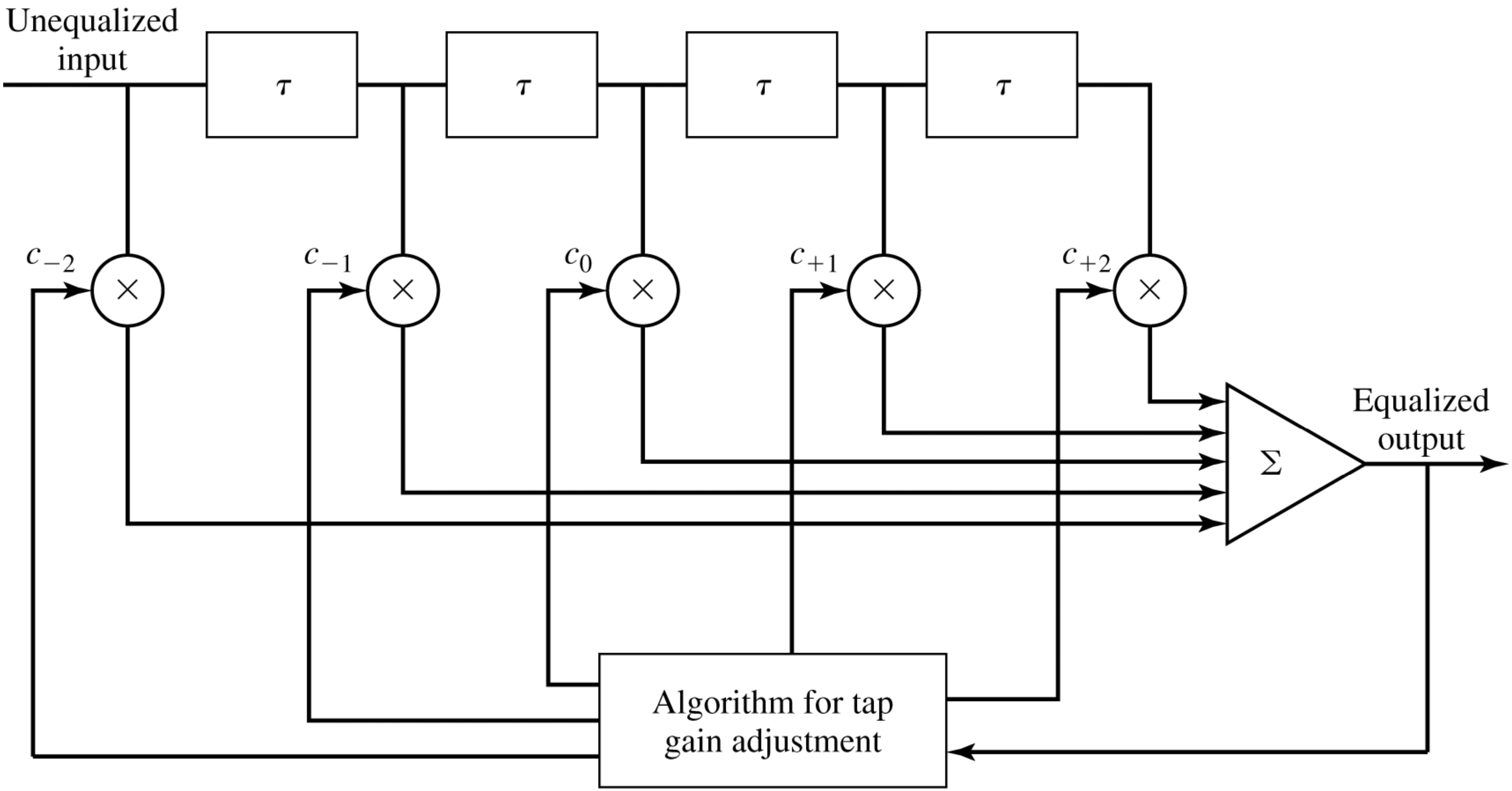


Figure 8.37

Linear transversal filter.

The time delay τ between asjacent taps may be selected as large as T, the symbol interval, in which case the FIR equalizer is called a **symbol-spaced equalizer**.

In this case the input to the equalizer is the sampled sequence given by (8.6.19).

However, we note that when $\frac{1}{T} < 2W$, frequencies in the received signal above the folding frequency $\frac{1}{T}$ are aliased into frequencies below $\frac{1}{T}$.

In this case, the equalizer compensates for the aliased channel-distorted signal.

On the other hand, when the time delay τ between adjacent taps is selected such that $\frac{1}{\tau} \geq 2W \geq \frac{1}{T}$, no aliasing occurs and, hence, the inverse channel equalizer compensates for the true channel distortion.

Since $\tau < T$, the channel equalizer is said to have **fractionally spaced taps** and it is called a **fractionally spaced equalizer**.

In practice, τ is often selected as $\tau = \frac{T}{2}$.

Notice that, in this case, the sampling rate at the output of the filter $G_{R(f)}$ is $\frac{2}{T}$.

The impulse response of the FIR equalizer is given by

$$g_E(t) = \sum_{n=-N}^{N} c_n \delta(t - n\tau) \tag{8.6.23}$$

and the corresponding frequency response of the FIR equalizer is given by

$$G_E(f) = \sum_{n=-N}^{N} c_n e^{-j2\pi f n\tau} \tag{8.6.24}$$

where $\{c_n\}$ the $(2N+1)$ equalizer coefficients, and N is chosen sufficiently large so that the equalizer spans the length of the ISI; i.e., $2N+1 \geq L$.

Since $X(f) = G_T(f)C(f)G_R(f)$ and $x(t)$ is the signal pulse corresponding to $X(f)$, then the equalized output signal pulse is given by

$$q(t) = \sum_{n=-N}^{N} c_n x(t - n\tau). \tag{8.6.25}$$

The zero-forcing condition can now be applied to the samples of $q(t)$ taken at times $t = mT$.

These samples are given by

$$q(mT) = \sum_{n=-N}^{N} c_n x(mT - n\tau), \quad m = 0, \pm 1, \cdots, \pm N. \tag{8.6.26}$$

Since there are $2N+1$ equalizer coefficients, we can control only $2N+1$ sampled values of $q(t)$. Specifically, we may force the conditions

$$q(mT) = \sum_{n=-N}^{N} c_n x(mT - n\tau)$$

$$= \begin{cases} 1, & m = 0, \\ 0, & m = \pm 1, \pm 2, \cdots, \pm N, \end{cases} \tag{8.6.27}$$

which may be expressed in matrix form as $\mathbf{X}_{\mathbf{c}} = \mathbf{q}$,

where $\mathbf{X}$ is a $(2N+1)\times(2N+1)$ matrix with elements $\{x(mT - n\tau)\}$,

$\mathbf{c}$ is the $(2N+1)$ coefficient vector, and $\mathbf{q}$ is the $(2N+1)$ column vector with one nonzero element.

Thus, we obtain a ste of $2N+1$ linear equations for the coefficients of the zero-forcing equalizer does not completely eliminate ISI because it has a finite length.

However, as $N \to \infty$, the ISI is completely eliminated.

Ex. 8.6.3

Consider a channel distorted pulse $x(t)$, at the input to the equalizer, given by the expression

$$x(t) = \frac{1}{1+\left(\frac{2t}{T}\right)^2}$$

where $\frac{1}{T}$ is the symbol rate.

The pulse is sampled at the rate $\frac{2}{T}$ and equalized by a zero-forcing equalizer.

Determine the coefficients of a five-tap zero-forcing equalizer.

Solution

According to (8.6.27), the zero-forcing equalizer must satisfy the equations

$$q(mT) = \sum_{n=-2}^{2} c_n x(mT - n\frac{T}{2})$$

$$= \begin{cases} 1, & m = 0, \\ 0, & m = \pm 1, \pm 2. \end{cases}$$

The matrix $\mathbf{X}$ with elements $x\left(mT - n \cdot \frac{T}{2}\right)$ is given as

$$\mathbf{X} = \begin{bmatrix} \frac{1}{5} & \frac{1}{10} & \frac{1}{17} & \frac{1}{26} & \frac{1}{37} \\ 1 & \frac{1}{2} & \frac{1}{5} & \frac{1}{10} & \frac{1}{17} \\ \frac{1}{5} & \frac{1}{2} & 1 & \frac{1}{2} & \frac{1}{5} \\ \frac{1}{17} & \frac{1}{10} & \frac{1}{5} & \frac{1}{2} & 1 \\ \frac{1}{37} & \frac{1}{26} & \frac{1}{17} & \frac{1}{10} & \frac{1}{5} \end{bmatrix}. \tag{8.6.28}$$

The corfficient vector $\mathbf{C}$ and the vector $\mathbf{q}$ are given as

$$\mathbf{g}_k\ \mathbf{c} = \begin{bmatrix} c_{-2} \\ c_{-1} \\ c_0 \\ c_1 \\ c_2 \end{bmatrix} \mathbf{q}$$

$$= \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}. \qquad (8.6.29)$$

Then, the linear equations $\mathbf{Xc} = \mathbf{q}$ can be solved by inverting the matrix $\mathbf{X}$.

Thus, we obtain

$$\mathbf{c}_{\mathbf{opt}} = \mathbf{X}^{-1}\mathbf{q}$$

$$= \begin{bmatrix} -2.2 \\ 4.9 \\ -3 \\ 4.9 \\ -2.2 \end{bmatrix}. \tag{8.6.30}$$

One drawback to the zero-forcing equalizer is that it ignores the presence of additive noise.

As a consequence, its use may result in significant noise enhancement.

This is easily seen by noting that in a frequency range where $C(f)$ is small, the channel equalizer $G_E(f) = \frac{1}{C(f)}$ compensates by placing a large gain in that frequency range.

Consequently, the noise in that frequency tange is greatly enhanced.

An alternative is to relax the zero ISI condition and select the channel equalizer characteristic such that the combined power in the residual ISI and the additive noise at the output of the equalizer is minimized.

A channel equalizer that is optimized based on the minimum mean-square-error (MMSE) criterion accomplishes the desired goal.

To elaborate, let us consider the noise-corrupted output of the FIR equalizer which is given by

$$z(t) = \sum_{n=-N}^{N} c_n y(t - n\tau) \tag{8.6.31}$$

where $y(t)$ is the input to the equalizer, given by (8.6.18).

The output is sampled at times $t = mT$.

Thus, we obtain

$$z(mT) = \sum_{n=-N}^{N} c_n y(mT - n\tau) . \tag{8.6.32}$$

The desired response sample at the output of the equalizer at $t = mT$ is the transmitted symbol a_m is given by

$$\text{MSE} = E[z(mT) - a_m]^2$$

$$= E\left[\sum_{n=-N}^{N} c_n y(mT - n\tau) - a_m\right]^2$$

$$= \sum_{n=-N}^{N}\sum_{k=-N}^{N} c_n c_k R_Y(n-k) - 2\sum_{k=-N}^{N} c_k R_{AY}(k) + E[a_m^2] \quad (8.6.33)$$

where the correlations are defined as

$$R_Y(n-k) = E[y(mT - n\tau)y(mT - k\tau)]$$

$$R_{AY} = E[y(mT - k_\tau)a_m] \quad (8.6.34)$$

and the expectation is taken with respect to the random information sequence $\{a_m\}$ and the additive noise.

The MMSE solution is obtained by differentiating (8.6.33) with respect to the equalizer coefficients $\{c_n\}$.

Thus, we obtain the necessary conditions for the MMSE as

$$\sum_{n=-N}^{N} c_n R_Y(n-k) = R_{YA}(k), \qquad k = 0, \pm 1, \pm 2, \cdots, \pm N\,. \quad (8.6.35)$$

These are $(2N+1)$ linear equations for the equalizer coefficients.

In contrast to the zero-forcing solution described previously, these equations depend on the atatistical properties (the autocorrelation function) of the noise as well as the ISI through the autocorrelation function $R_Y(n)$.

In practice, we would not normally know the autocorrelation function $R_{AY}(n)$.

However, these correlation sequences can be estimated by transmittinf a test signal over the channel and using the time average estimates

$$\hat{R}_Y(n) = \frac{1}{K}\sum_{k=1}^{K} y(kT - n\tau)\,y(kT)$$

$$\hat{R}_{AY}(n) = \frac{1}{K}\sum_{k=1}^{K} y(kT - n\tau)a_k \ . \tag{8.6.36}$$

In place of the ensemble averages to solve for the equalizer coefficients given by (8.6.35).

Adaptive Equalizers

We have shown that the tap coefficients of a linear equalizer can be determined by solving a set of linear equations.

In the zero-forcing optimization criterion, the linear equations are given by (8.6.27).

On the other hand, if the optimization criterion is based on minimizing the MSE, the optimum equalizer coefficients are determined by solving the set of linear equations given by (8.6.35).

In both cases, we may express the set of linear equations in the general matrix form

$$\mathbf{B}\mathbf{c} = \mathbf{d} \tag{8.6.37}$$

where $\mathbf{B}$ is a $(2N+1)\times(2N+1)$ matrix,

$\mathbf{c}$ is a column vector representing the $2N+1$ equalizer coefficients, and

$\mathbf{d}$ is a $(2N+1)$-dimensional column vector.

The solution of (8.6.37) yields

$$\mathbf{c}_{\text{opt}} = \mathbf{B}^{-1}\mathbf{d}\,. \tag{8.6.38}$$

In practical implementations of equalizers, the solution of (8.6.37) for the optimum coefficient vector is usually obtained by an iterative procedure that avoids the explicit computation of the inverse of the matrix $\mathbf{B}$.

The simplest iterative procedure is the method of steepest descent, in which one begins by choosing arbitrarily the coefficient vector $\mathbf{c}$, say $\mathbf{c}_0$.

This initial choice of coefficients corresponds to a point on the criterion function that is begin optimized.

For example, in the case of the MSE criterion, the initial guess $\mathbf{c}_0$ corresponds to a point on the quadratic MSE surface in the $(2N+1)$ - dimensional space of coefficients.

The gradient vector, defined as g_0, which is the derivative of the MSE with respect to the $2N+1$ filter coefficients, is then computed at this point on the criterion surface and each tap coefficient is changed in the direction opposite to its corresponding gradient component.

The change in the j th tap coefficient is proportional to the size of the j th gradient component.

For example, the gradient vector, denoted as $\mathbf{g}_k$, for the MSE criterion, found by taking the derivatives of the MSE with respect to each of the $2N+1$ coefficients, is given by

$$\mathbf{g}_k = \mathbf{B}\mathbf{c}_k - \mathbf{d}, \qquad k = 0, 1, 2, \cdots \tag{8.6.39}$$

Then the coefficient vector $\mathbf{c}_k$ is updated according to the relation

$$\mathbf{c}_{k+1} = \mathbf{c}_k - \Delta \mathbf{g}_k \tag{8.6.40}$$

where Δ is the **step-size parameter** for the iterative procedure.

To ensure convergence of the iterative procedure, Δ is chosen to be a small positive number.

In such a case, the gradient vector $\mathbf{g}_k$ converges toward zero; i.e., $\mathbf{g}_k \to \mathbf{0}$ as $k \to \infty$, and the coefficient vector $\mathbf{c}_k \to \mathbf{c}_{\text{opt}}$ as shown in Figure 8.38 based on two-dimensional optimization.

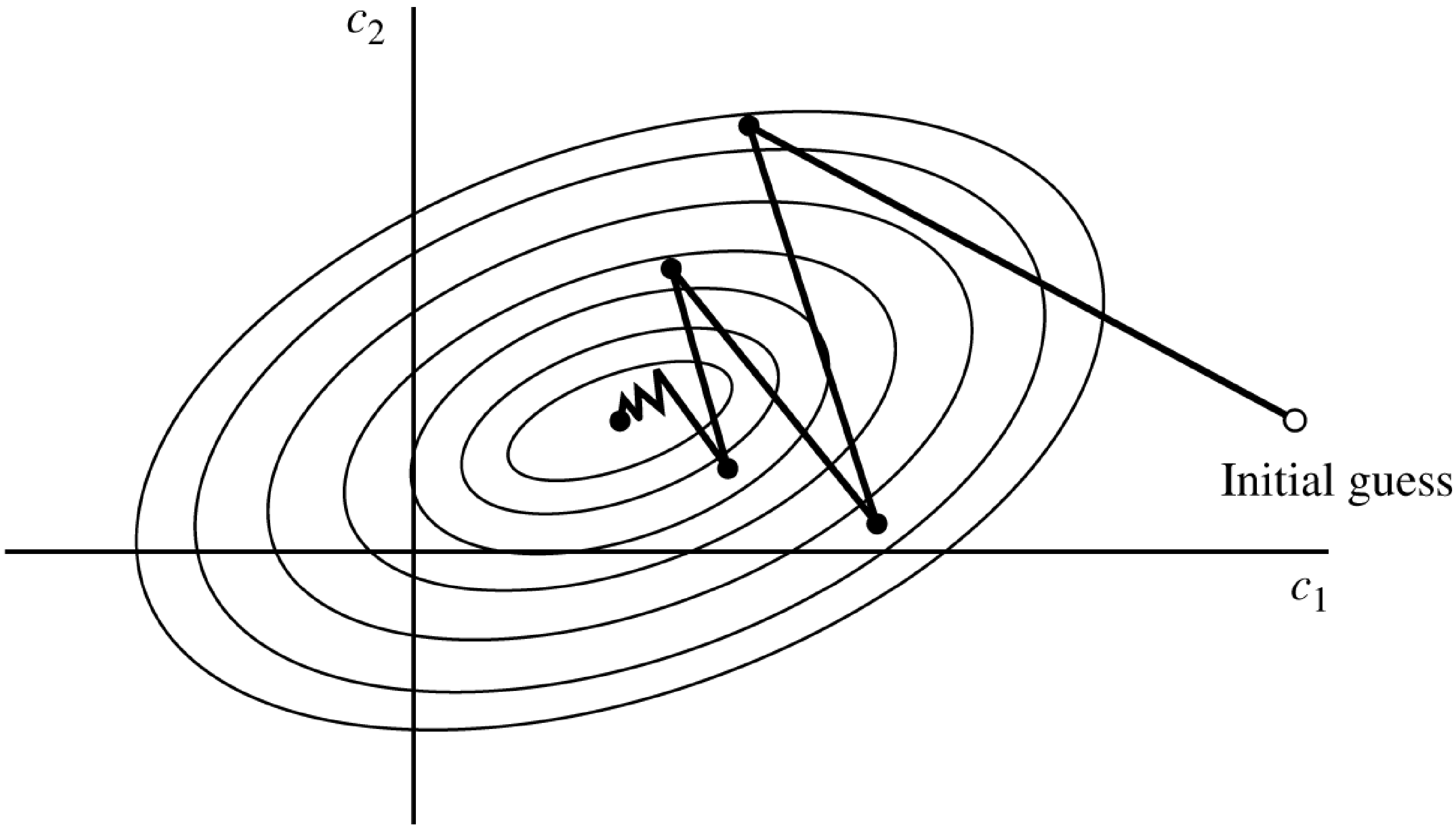


Figure 8.38

Example of convergence characteristics of a gradient algorithm.
(From *Introduction to Adaptive Arrays*, by R.A. Monzigo and T.W. Miller; © 1980 by John Wiley & Sons. Reprinted with permission of the publisher.)

In general, convergence of the equalizer tap coefficients to $\mathbf{c}_{\text{opt}}$ corresponds to a fraction of a second.

Adaptive channel equalization is required for channels whose characteristics change with time. In such a case, ths ISI varies with time.

The channel equalizer must track such time variations in the channel response and adapt its coefficients to reduce the ISI.

In the context of the above discussion, the optimum coefficients vector $\mathbf{c}_{\text{opt}}$ varies with time due ti time variations in the matrix $\mathbf{B}$ and, for the case of the MSE criterion, time variations in the vector $\mathbf{d}$.

Under these conditions, the iterative method described above can be modified to use estimates of the gradient components.

Thus, the algorithm for adjusting the equalizer tap coefficients may be expressed as

$$\hat{\mathbf{c}}_{k+1} = \hat{\mathbf{c}}_k - \Delta \hat{\mathbf{g}}_k \tag{8.6.41}$$

where $\hat{\mathbf{g}}_k$ denotes an estimate of the gradient vector $\mathbf{g}_k$ and

$\hat{\mathbf{c}}_k$ denotes the estimate of the tap coefficient vector.

In the case of the MSE criterion, the gradient vector $\mathbf{g}_k$ given by (8.6.39) may also be expressed as (see Problem 8.46).

$$\mathbf{g}_k = -E(e_k \mathbf{y}_{k)}.$$

An estimate $\hat{\mathbf{g}}_k$ of the gradient vecrot at the kth iteration is computed as

$$\hat{\mathbf{g}}_k = -e_k \mathbf{y}_k \tag{8.6.42}$$

where e_k denotes the difference between the desired output from the equalizer at the kth time instant and the actual output $z(kT)$,

and $\mathbf{y}_k$ denotes the column vector of $2N+1$ received signal values contained in the equalizer at time instant k.

The error signal e_k is expressed as

$$e_k = a_k - z_k \tag{8.6.43}$$

where $z_k = z(kT)$ is the equalizer output given by (8.6.32) and a_k is the desired symbol.

Hence, by substituting (8.6.42) into (8.6.41), we obtain the adaptive algorithm for optimizing the tap coefficints (based on the MSE criterion) as

$$\hat{\mathbf{c}}_{k+1} = \hat{\mathbf{c}}_k + \Delta e_k \mathbf{y}_k \,. \tag{8.6.44}$$

Since an estimate of the gradient vector is used in (8.6.44) the algorithm is called a **stochastic gradient algorithm** or the LMS algorithm.

A block diagram of an adaptive equalizer that adapts its tap coefficients according to (8.6.44) is shown in Figure 8.39.

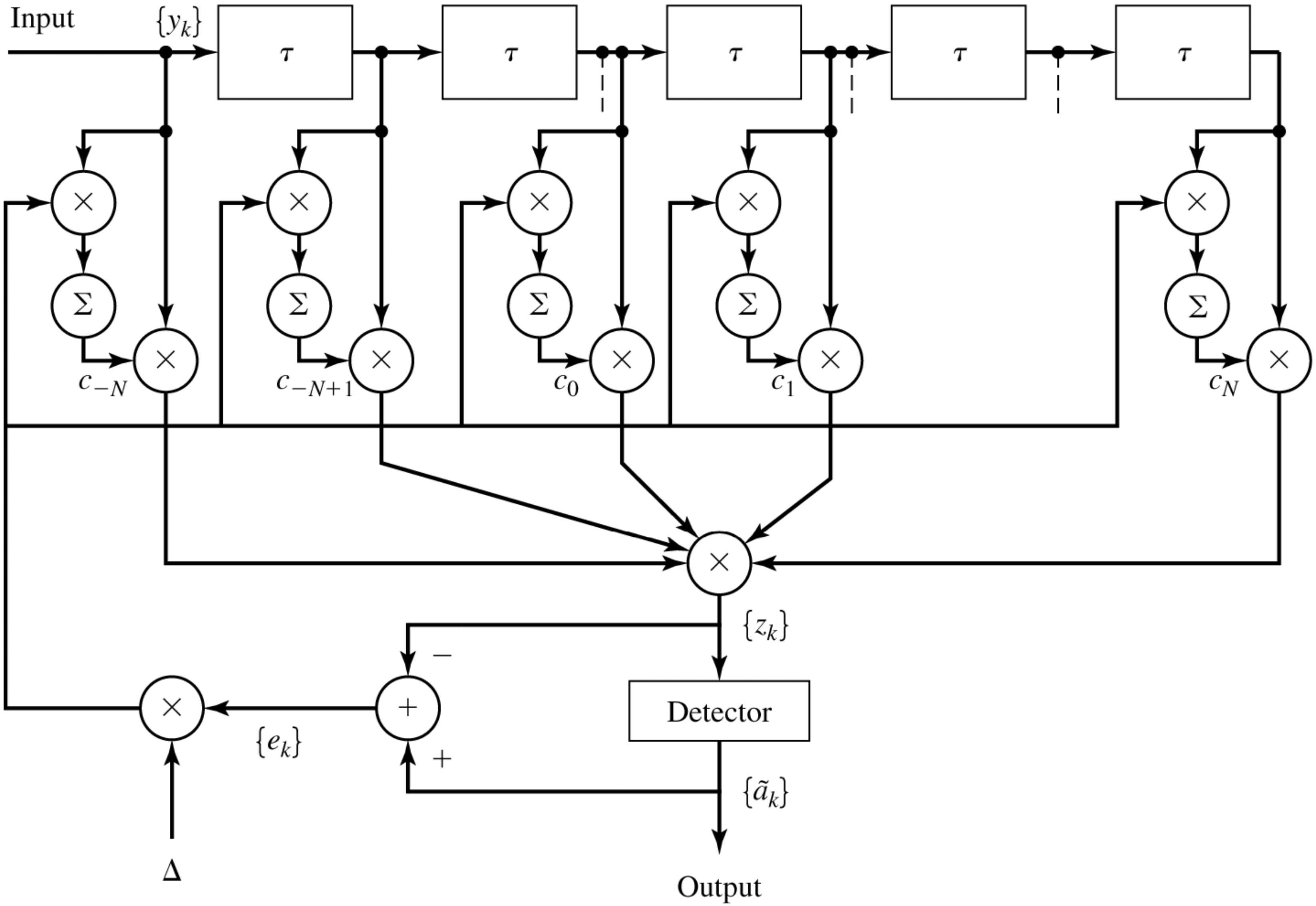


Figure 8.39

Linear adaptive equalizer based on the MSE criterion.

Note that the difference between the desired output a_k and the actual output z_k from the equalizer is used to form the error signal e_k multiplies the received signal values $\{y(kT - n\tau)\}$ at the $2N+1$ taps.

The products $\Delta e_k y(kT - n\tau)$ at the $(2N+1)$ taps are then added to the previous values of the tap coefficients to obtain the updated tap coefficients, according to (8.6.44).

This computation is repeated for each received symbol.

Thus, the equalizer coefficients are updated at the symbol rate.

Initially, the adaptive equalizer is trained by the transmission of a known pseudo-random sequence $\{a_m\}$ over the channel.

At the demodulator, the equalizer employs the known sequence to adjust its coefficients.

Upon initial adjustment, the adaptive equalizer switches from a **traning mode** to a **decision-directed mode**, in which case the decisions at the output of the detector are sufficiently reliable so that the error signal is formed by computing the difference between the detector output and the equalizer output; i.e.,

$$e_k = \tilde{a}_k - z_k \qquad (8.6.45)$$

where $\tilde{a}_k$ is the output of the detector.

In general, decision erroes at the output of the detector occur infrequently and, consequently, such errors have little effect on the performance of the tracking algorithm given by (8.6.44).

A rule of thumb for selecting the step-size parameter so as to ensure convergence and good tracking capabilities in slowly varying channel is given by

$$\Delta = \frac{1}{5(2N+1)P_R} \qquad (8.6.46)$$

where P_R denotes the received signal-plus-noise power, which can be estimated from the received signal.

The convergence characteristics of the stochastic gradient algorithm in (8.6.44) is shown in Figure 8.40.

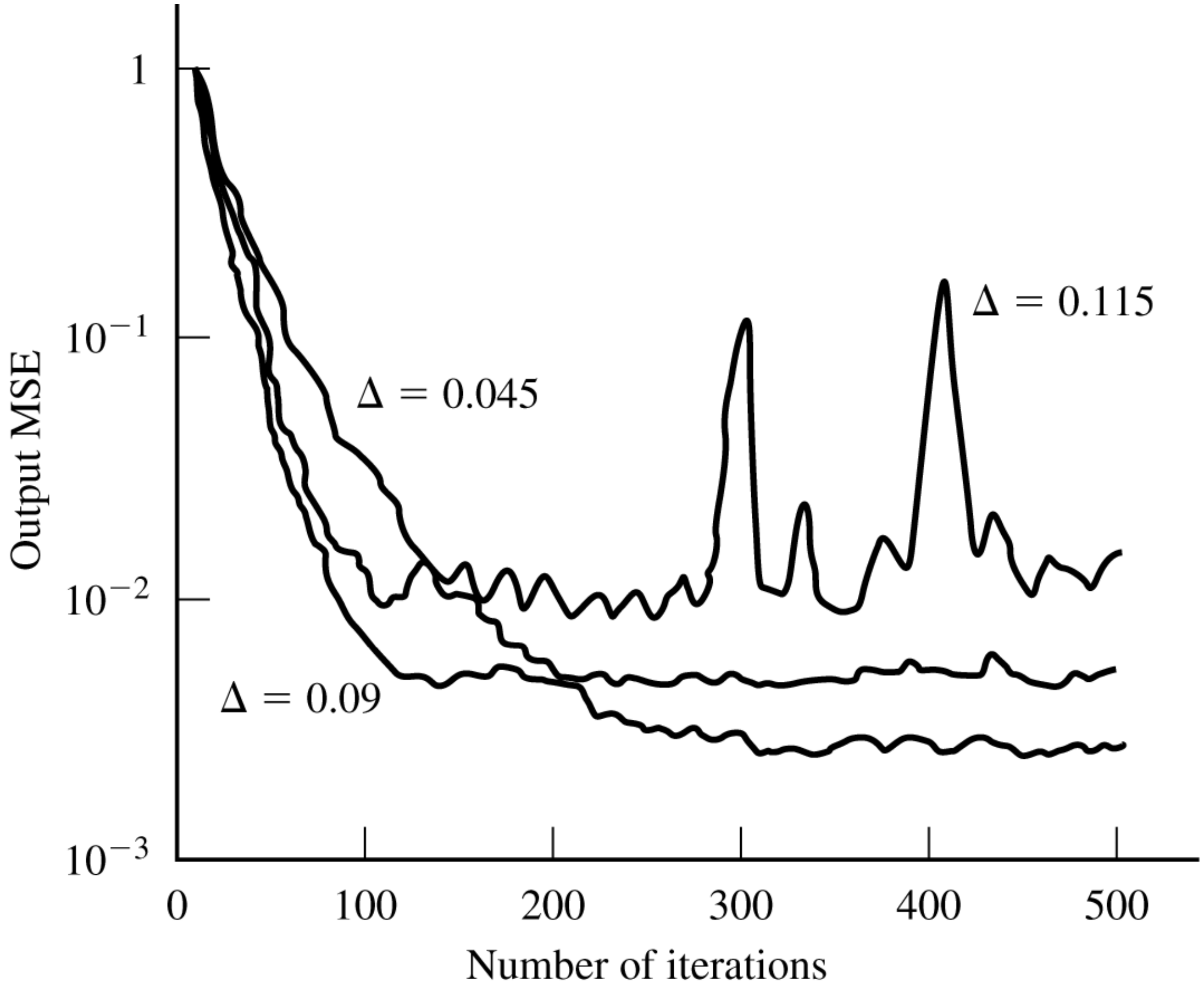


Figure 8.40

Initial convergence characteristics of the LMS algorithm with different step sizes.
(From *Digital Signal Processing* by J. G. Proakis and D. G.Manolakis; © 1988, Macmillan. Reprinted with permission of the publisher.)

These graphs were obtained from a computer simulation of an 11-tap adaptive equalizer operating a channel with a rather modest amount of ISI.

The input signal-plus-noise power P_R was normalized to unity.

The rule of thumb given in (8.6.46) for selecting the step size gives $\Delta = 0.018$.

The effect of making Δ too large is illustrated by the large jumps in MSE as shown for $\Delta = 0.115$.

As Δ is decreased, the convergence is slowed somewhat, but a lower MSE is achieved, indicating that the estimated coefficients are closer to $\mathbf{c}_{\text{opt}}$.

Although we have described in some detail the operation of an adaptive equalizer which is optimized on the basis of the MSE criterion, the operation of an adaptive equalizer based on the zero-forcing method is very similar.

The major difference lies in the method for estimating the gradient vectors $\mathbf{g}_k$ at each iteration.

A block diagram of an adaptive zero-forcing equalizer is shown in Figure 8.41.

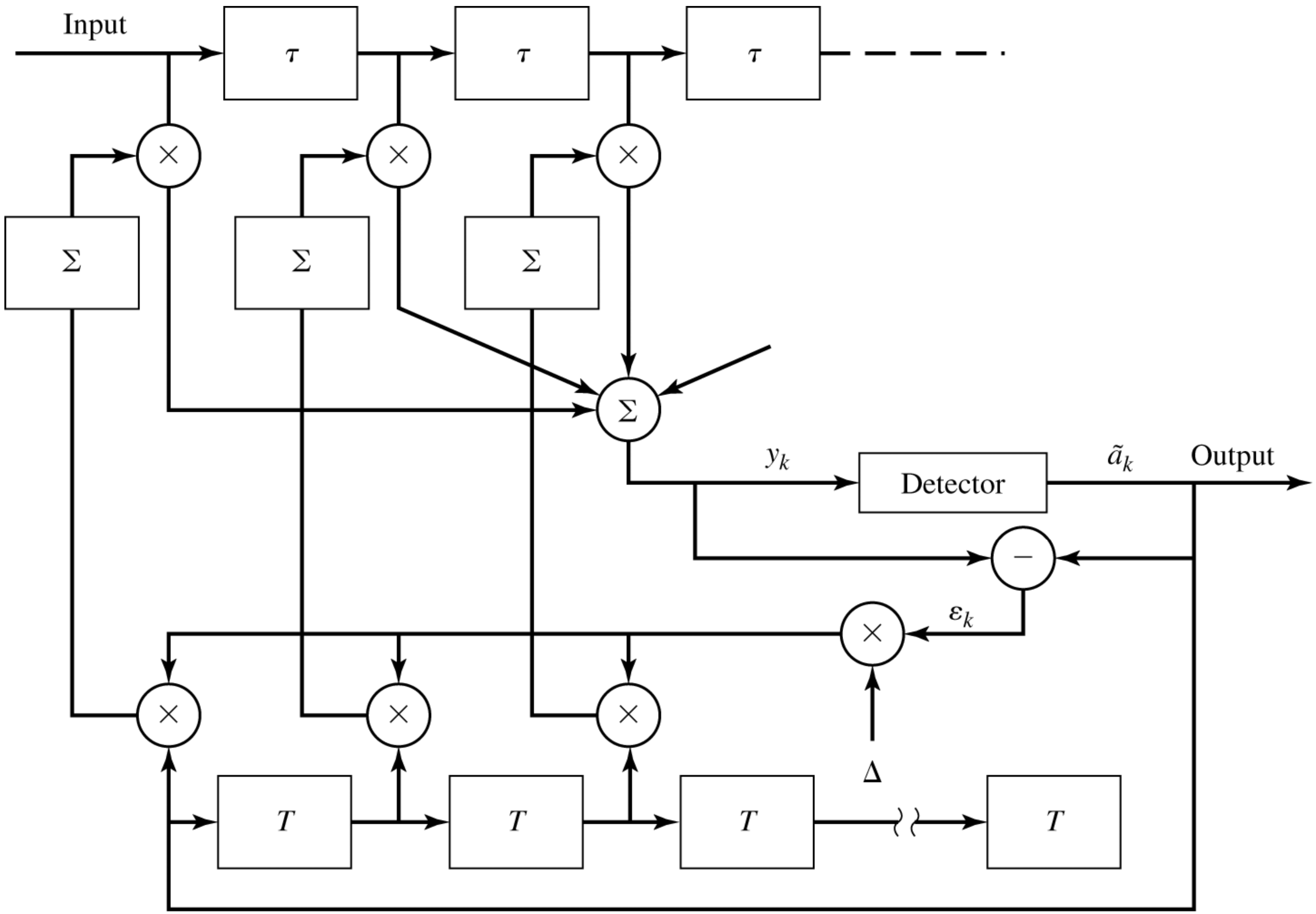


Figure 8.41

An adaptive zero-forcing equalizer.

Decision-Feedback Equalizer

The linear filter equalizers described above are very effective on channels, such as wireline telephone channels, where the ISI is not severe.

The severity of the ISI is directly related to the spectral characteristics and not necessarily to the time span of the ISI.

For example, consider the ISI resulting from two channels which are shown in Figure 8.42.

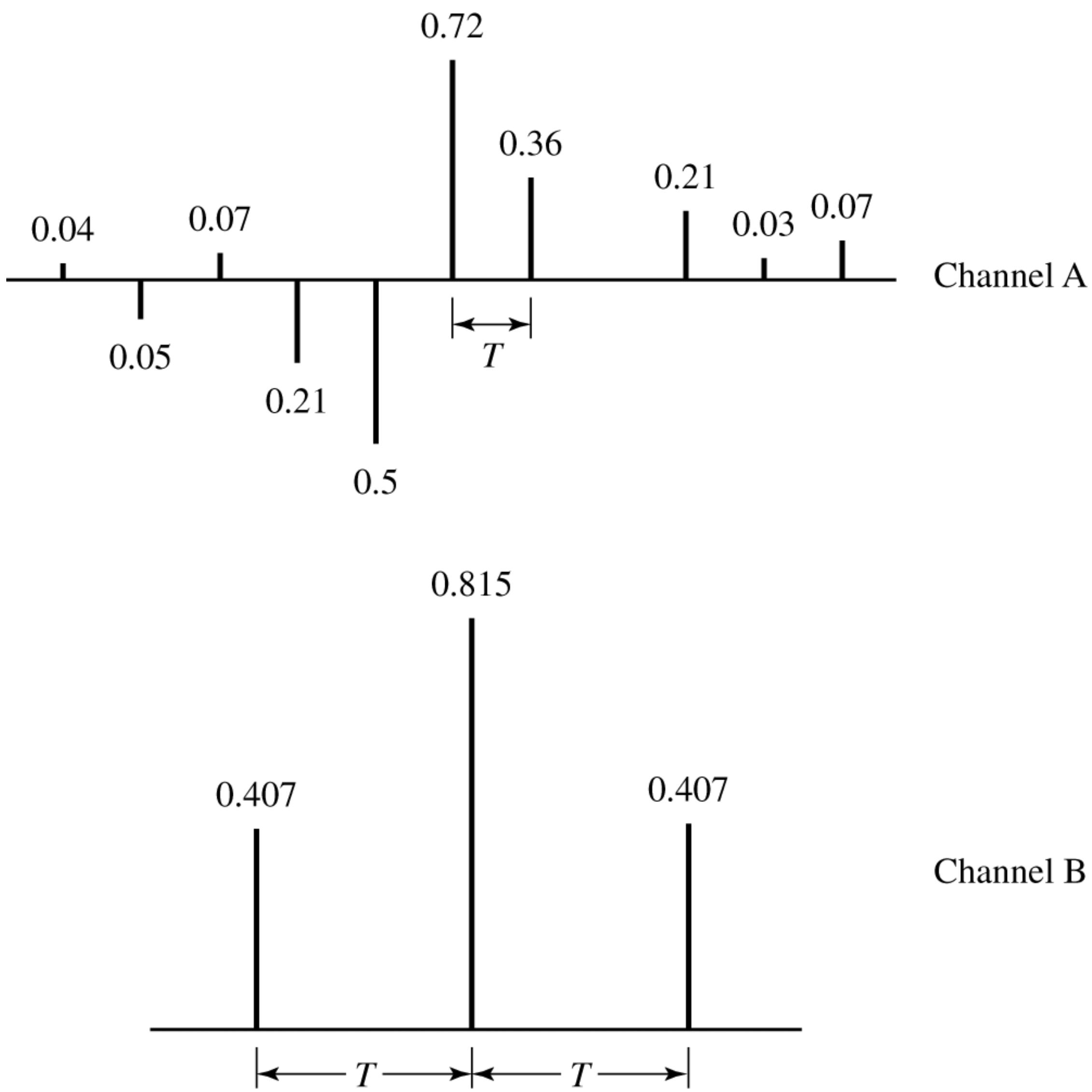


Figure 8.42

Two channels with ISI.

The time span for the ISI in Channel A is 5 symbol intervals on each side of the desired signal component, which has a value of 0.72.

On the other hand, the time span for the ISI in Channel B is one symbol interval on each side of the desired signal component, which has a value of 0.815.

The energy of the total response is normalized to unity for both channels.

In spite of the shorter ISI span, Channel B results in more severe ISI.

This is evidenced in the frequency response characteristics of these channels, which are shown in Figure 8.43.

We observe that Channel B has a spectral null (the frequency response characteristics of these channels, which are shown in Figure 8.43.

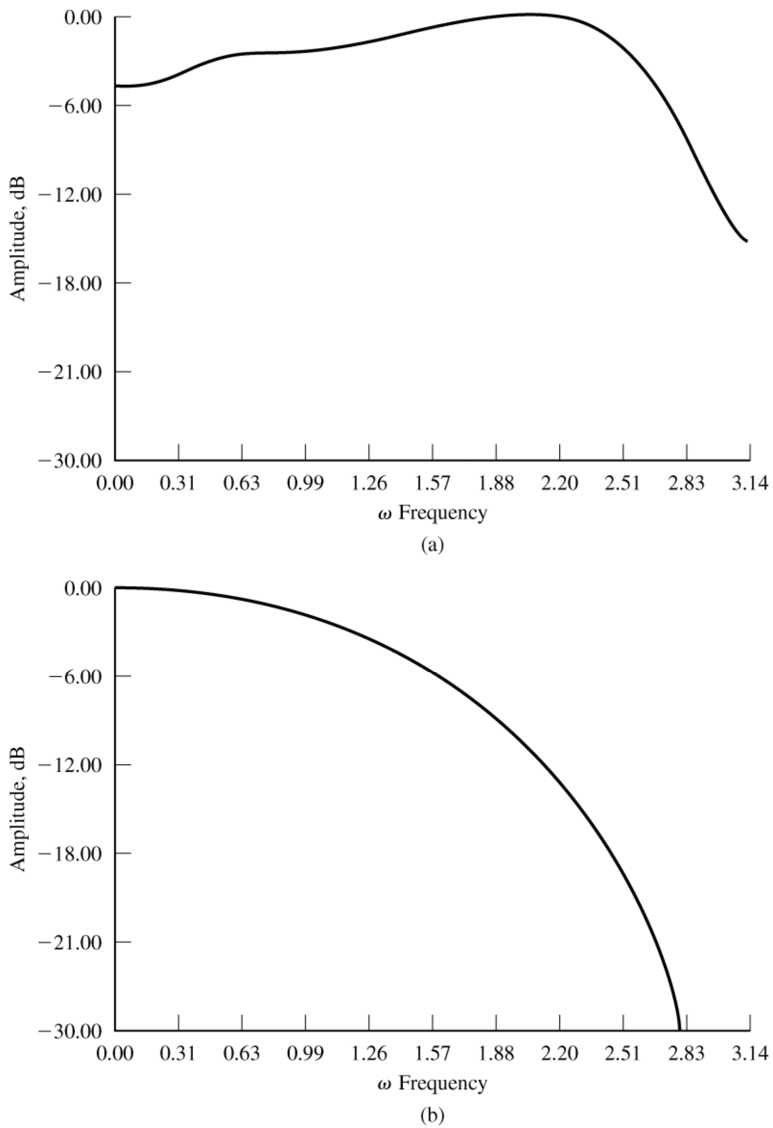


Figure 8.43

Amplitude spectra for (a) channel A shown in Figure 8.42(a) and (b) channel B shown in Figure 8.42(b).

We observe that Channel B has a spectral null (the frequency response $C(f)=0$ for some frequencies in the band $|f|\le W$) at $f=\frac{1}{2T}$, whereas this does not occur in the case of Channel A.

Consequently, a linear equalizer will introduce a large gain in its frequency response to compensate for the channel null.

Thus, the noise in Channel B will be enhanced much more than in Channel A.

This fact is borne out by the computer simulation results for the performance of the linear equalizer for the two channels, as shown in Figure 8.44.

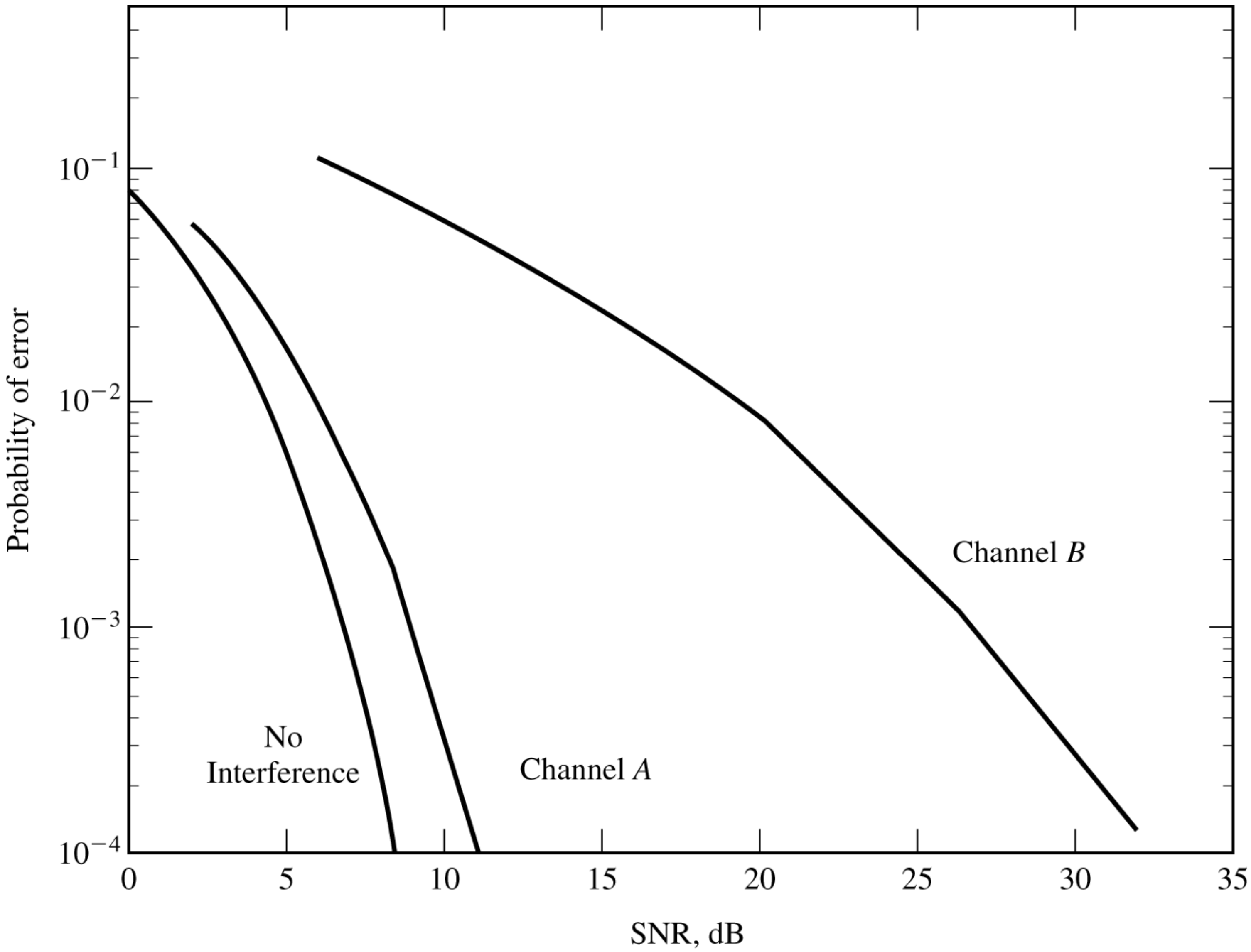


Figure 8.44

Error-rate performance of linear MSE equalizer.

Hence, the basic limitation of a linear equalizer is that it performs poorly on channels having spectral nulls.

Such channels are often encountered in radio channels, such as those used for cellular radio communications.

A **decision-feedback equalizer** (DFE) is a nonlinear equalizer that employs previous decision to eliminate the ISI caused by previously detected symbols on the current symbol to be detected.

A simple block diagram for a DFE is shown in Figure 8.45.

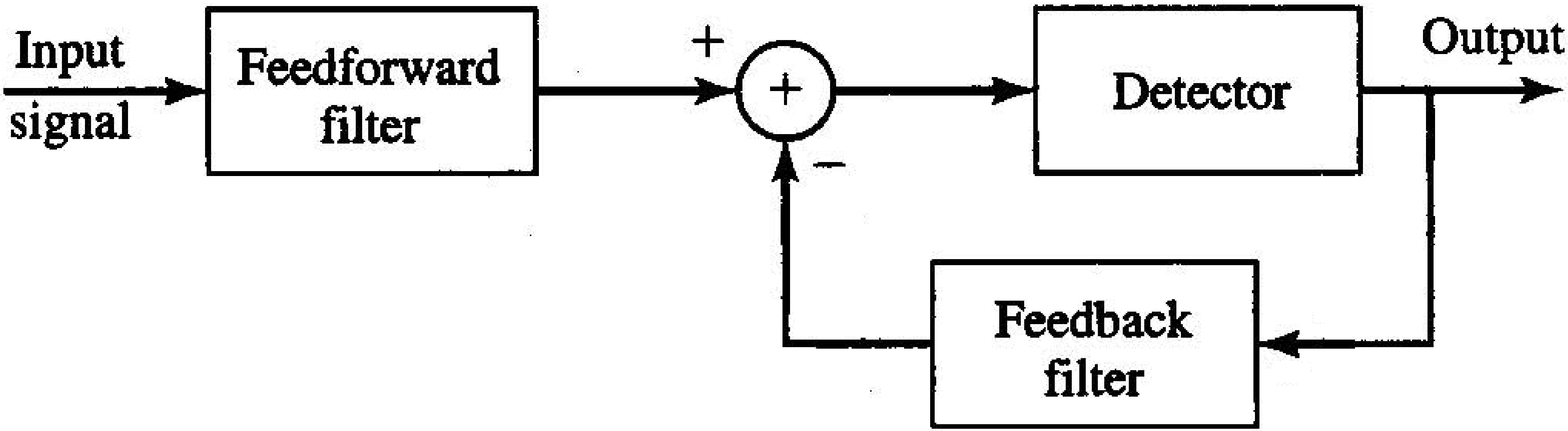


Figure 8.45 Block diagram of DFE.

The DFE consists of two filters. The first filter is called a **feedforward filter** and it is generally a fractionally spaced FIR filter with adjustable tap coefficients.

This filter is identical in form to the linear equalizer described above.

Its input is the received filtered signal $y(t)$.

The second filter is a **feedback filter**.

It is implemented as an FIR filter with symbol-spaced taps having adjustable coefficients.

Its input is the set of previously detected symbols.

The output of the feedback filter is subtracted from the output of the feedforward filter to form the input to the detector. Thus, we have

$$z_m = \sum_{n=1}^{N_1} c_n y(mT - n\tau) - \sum_{n=1}^{N_2} b_n \tilde{a}_{m-n} \tag{8.6.47}$$

where $\{c_n\}$ and $\{b_n\}$ are the adjustable coefficients of the feedforward and feedback filters, respectively,

$\tilde{a}_{m-n}$, $n = 1, 2, \cdots, N_2$, are the previously detected symbols,

N_1 is the length of the feedforward filter, and

N_2 is the length of the feedback filter.

Based on the input z_m, the detector determines which of the possible transmitted symbol is closest in distance to the input signal z_m.

Thus, it makes its decision and output $\tilde{a}_m$.

What makes the DFE nonlinear is the nonlinear characteristic of the detector which provides the input to the feedback filter.

The tap coefficient of the feedforward and feedback filters are selected to optimize some desired performance measure.

For mathematical simplicity, the MSE criterion is usually applied and a stochastic gradient algorithm is commonly used to implement an adaptive DFE.

Figure 8.46 shows the block diagram of an adaptive DFE whose tap coefficients are adjusted by means of the LMS stochastic gradient algorithm.

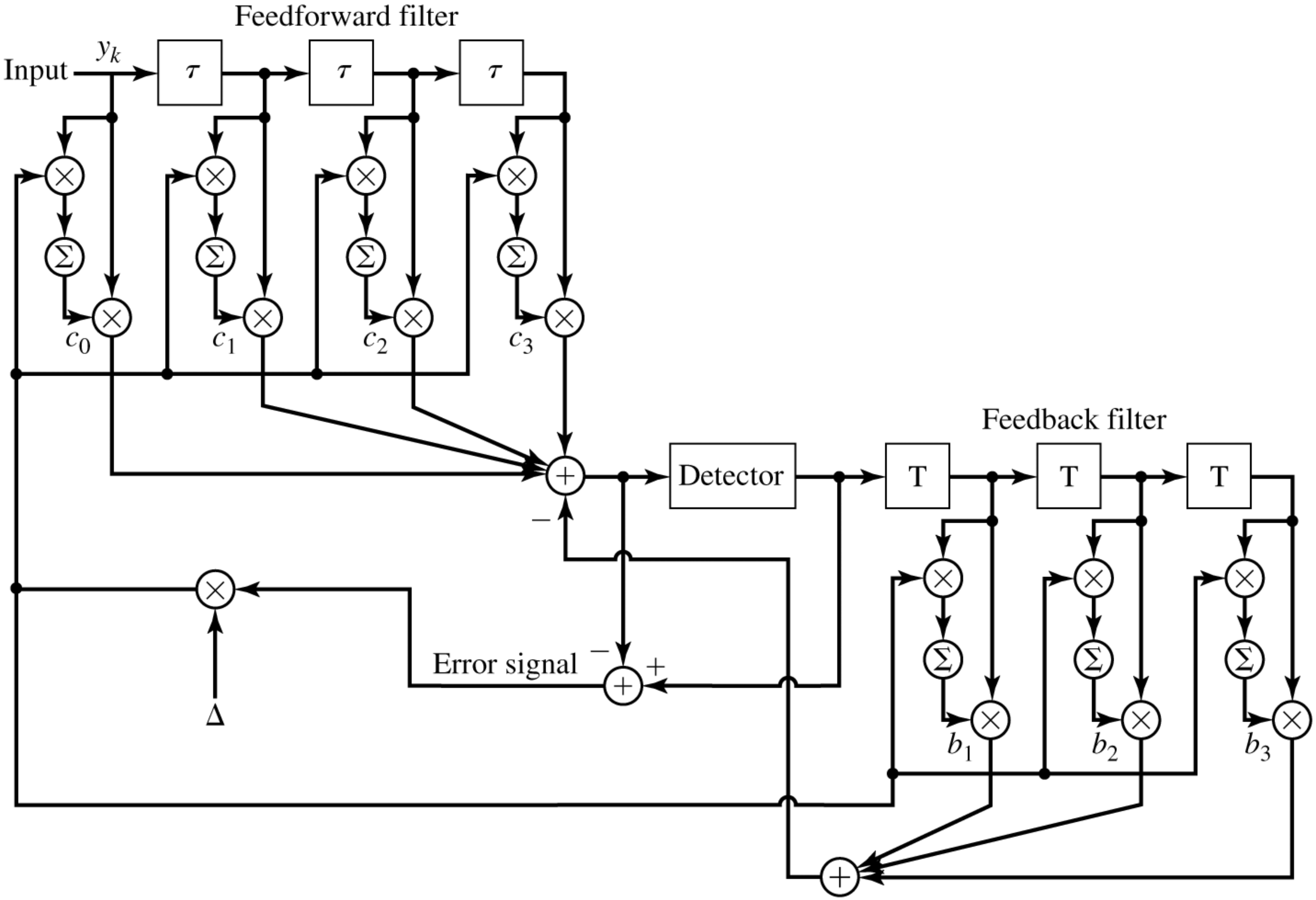


Figure 8.46

Adaptive DFE.

Figure 8.47 shows the probability of error performance of the DFE, obtained by computer simulation, for binary PAM transmission over Channel B.

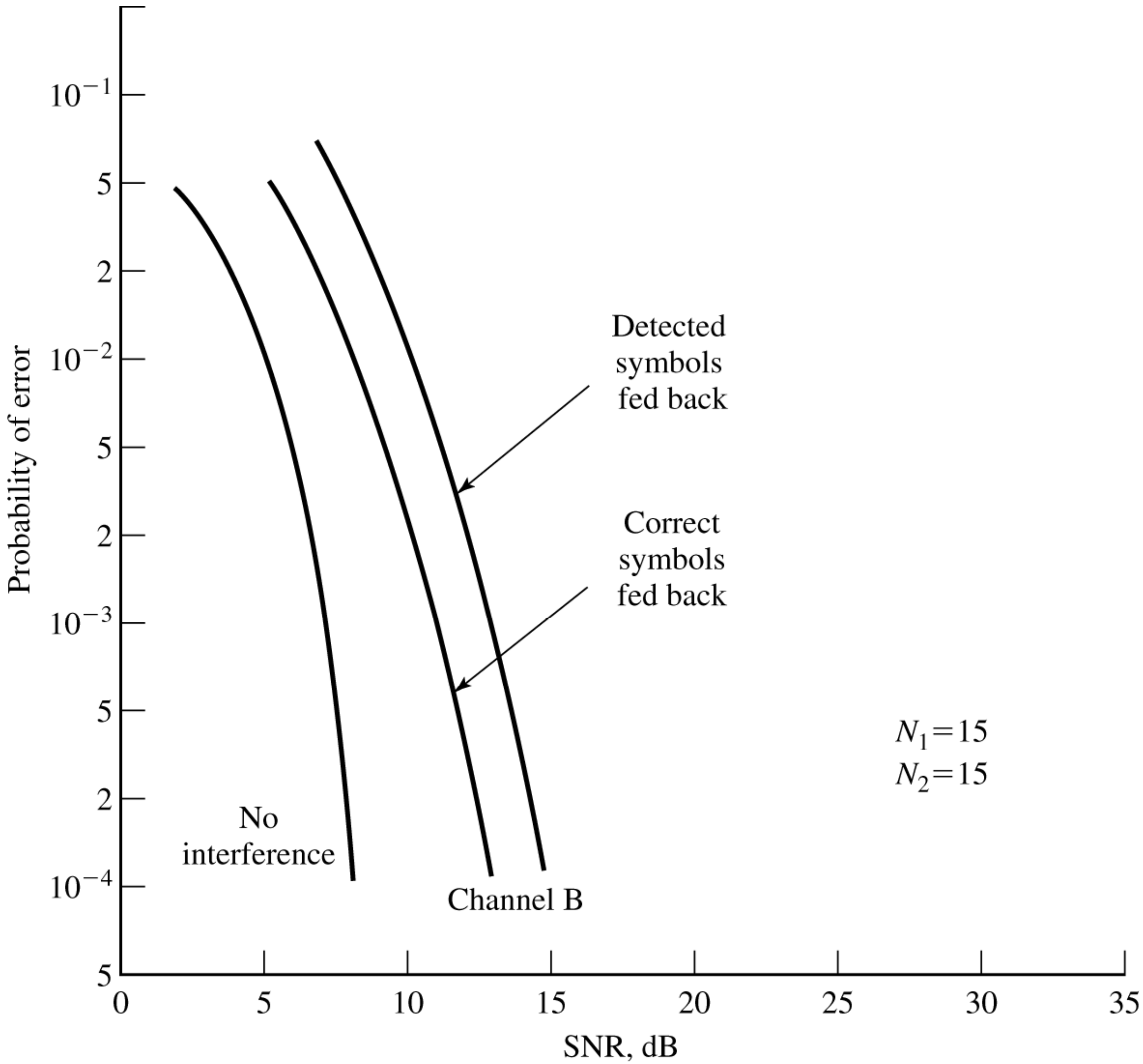


Figure 8.47

Performance of DFE with and without error propagation.

The gain in performance relative to that of a linear equalizer is clearly evident.

We should mention that decision errors from the detector that are fed to the feedback filter have a small effect on the performance of the DFE.

In general, a small loss in performance of one to two dB is possible at error rates below 10^{-2}, but the decision errors in the feedback filters are not catastrophic.

Although the DFE outperforms a linear equalizer, it is not the optimum equalizer freom the viewpoint of minimizing the probability of error.

As indicated previously equalizer from the optimum detector in a digital communication system in the presence of ISI is a ML symbol sequence detector.

It is particularly appropriate for channels with severe ISI, when the ISI spans only a few signals.

For example Figure 8.48 shows the error probability performance of the Viterbi algorithm for binary PAM signal transmitted through channel B (see Figure 8.42).

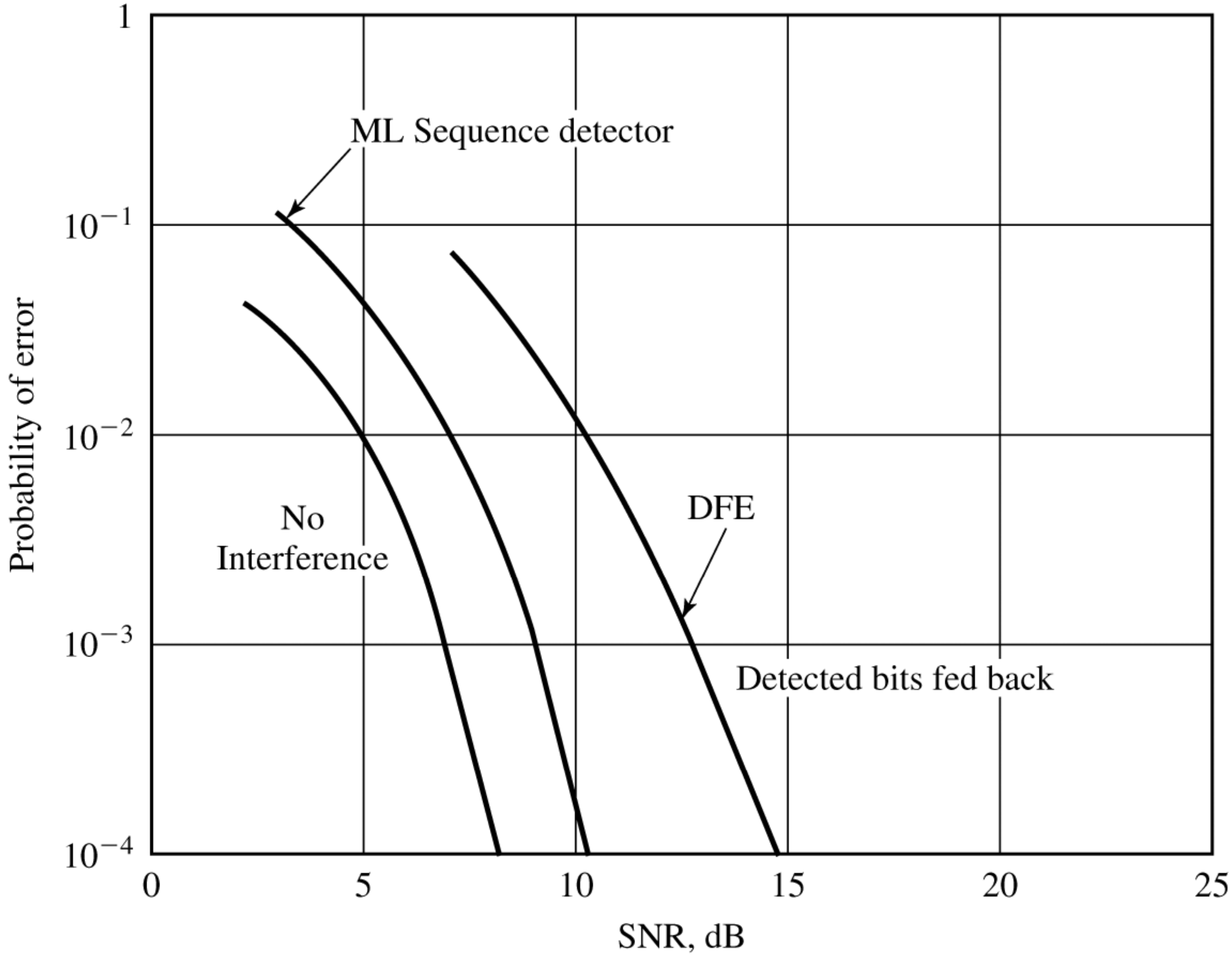


Figure 8.48

Performance of Viterbi detector and DFE for channel B.

For purposes of comparison, we also illustrate the probability of error for a decision feedback equalizer.

Both results were obtained by computer simulation.

We observe that the performance of the ML sequence detector is about 4.5-dB better than that of the DFE at an error probability of 10^{-4}.

Hence, this is one example where the ML sequence detector provides a significant performance gain on a channel which has a relatively short ISI span.

In conclusion, we mention that adaptive equalizers are widely used in high-speed digital communication systems for telephone channels.

High-speed telephone line modems (bit rate above 2400 bps) generally include an adaptive equalizer that is implemented as an FIR filter with coefficients that are adjusted based on the MMSE criterion.

Depending on the data speed, the equalizer typically soans between 20 and 70 symbols.

The LMS algjrithm given by (8.6.44) is usually employed for the adjustment of the equalizer coefficients adaptively.

8.7 Multicarrier Modulation And OFDM

We observed that inter-symbol interference results in performance degradation, even when the optimum detector is used to recover the information symbols at the receiver.

An alternative approach to designing a bandwidth-efficient communication system in the presence of channel distortion is to subdivide the available channel bandwidth into a number of equal-bandwidth subchannels each of which bandwidth is sufficiently narrow so that the frequency response of the subchannels are nearly ideal.

Such a subdivision of the overall bandwidth into smaller subchannels with bandwidth Δf is shown in Figure 8.49.

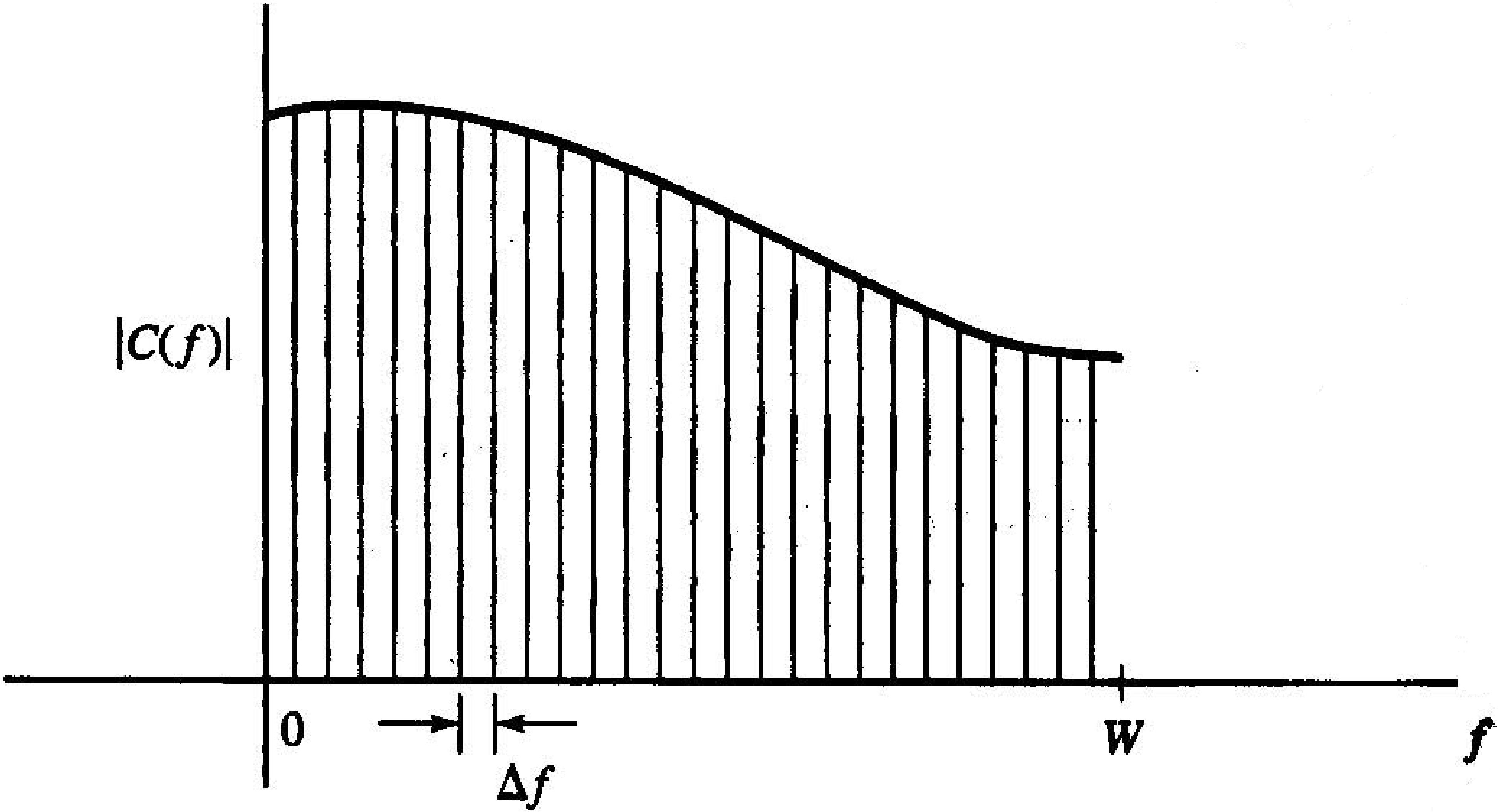


Figure 8.49 Subdivision of the channel bandwidth W into narrowband subchannels of equal width Δf.

Then, different information symbols are transmitted simultaneously through $K = \frac{W}{\Delta f}$ subchannels.

Consequently, the data is transmitted by frequency-division multiplexing (FDM).

With each subchannel, we associate a carrier signal given by

$$x_k(t) = \sin 2\pi f_k t, \quad k = 0, 1, \cdots, K-1, \tag{8.7.1}$$

where f_k is the mid-frequency of the k th subchannel.

By selecting the symbol rate $\frac{1}{T}$ on each of the subchannels to be equal to the separation Δf between adjacent subcarriers, the subcarriers become orthogonal over the symbol interval T, independent of the relative phase relationship between subcarriers.

That is,

$$\int_0^T \sin(2\pi f_k t + \phi_k) \sin(2\pi f_j t + \phi_j) dt = 0 \tag{8.7.2}$$

where $f_k - f_j = \frac{n}{T}, \quad n = 1, 2, \cdots,$ independent of the phase ϕ_k and ϕ_j.

In this case, we have orthogonal frequency-division multiplexing (OFDM).

With an OFDM system having K subchannels, the symbol rate on each subcarrier is reduced by a factor of K relative to the symbol rate on a single carrier system which employs the entire bandwidth W to transmit data at the same rate as OFDM.

Hence, the symbol interval in the OFDM system is $T = KT_S$, where T_S is the symbol interval in the single-carrier system.

By selecting K to be sufficiently large, the symbol interval T can be made significantly larger than the duration of time dispersion of the channel.

Thus, intersymbol interference can be made arbitrarily small by the selection of K.

In other words, each subchannel appears to have a fixed frequency response $C(f_k), \quad k = 0, 1, \cdots, K-1$.

As long as we maintain time synchronization among the subcarriers, OFDM allows us to transmit a different number of bits/symbol on each subcarrier.

Hence, subcarriers which yield a higher SNR due to a lower attenuation can be modulated by a higher level modulation scheme (with more bits/symbol) than subchannels which yield a lower SNR (due to higher attenuation).

For example, QAM with various constellation sizes may be used in an OFDM system.

The modulator and demodulator in an OFDM system can be implemented by use of a parallel bank of filters based on the diecrete Fourier transform (DFT).

When the number of subchannels is large, say $K > 25$, the modulator and demodulator in an OFDM system are efficiently implemented by using the fast Fourier transform algorithm (FFT) to compute the DFT.

Next, we describe an OFDM system in which the modulator and demodulator are implemented based on the DFT.

A major problem with the multicarrier modulation in general and OFDM system in particular is the high peak-to-average power ratio (PAR) that is inherent in the transmitted signal.

Large signal peaks occur in the transmitted signal when the signals in the K subchannels add constructively in phase.

Such large signal peaks may saturate the power amplifier at the transmitter to cause intermodulation distortion in the transmitted signal.

Intermodulation distortion can be reduced and avoided by reducing the average power of the transmitted signal to make the power amplifier to operate in the linear range.

Such power reduction results in inefficient operation of the OFDM system.

A variety of methods have been devised to reduce PAR in multicarrier systems.

A relatively simple method is to insert different phase shifts in each of the subcarriers, where the phase shifts are selected pseudorandomly, or by means of some algorithm, to reduce the PAR.

8.7.1 An OFDM System Implemented via the FFT Algorithm

The basic block diagram of the OFDM is shown in Figure 8.50.

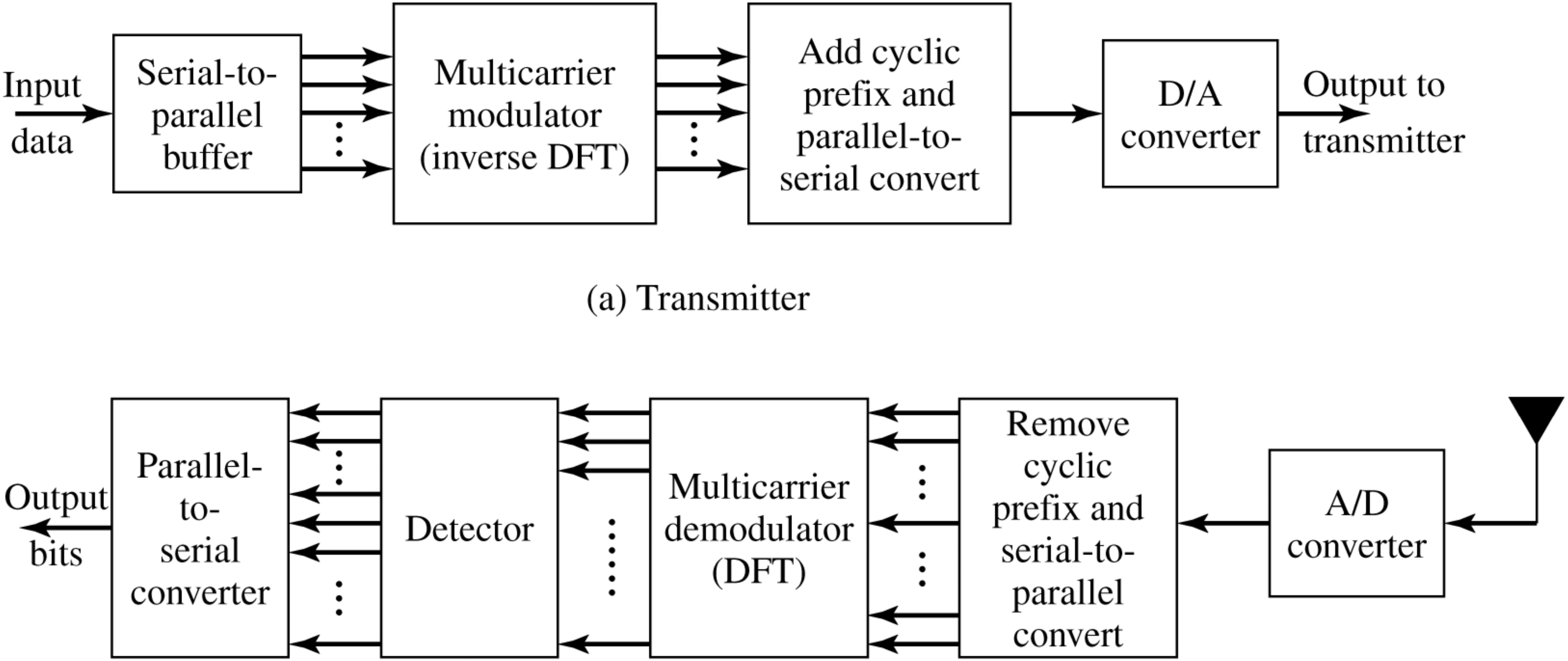


Figure 8.50

Block diagram of a multicarrier OFDM digital communication system.

A serial-to-parallel buffer subdivides the information sequence into frames of B_f bits.

The B_f bits in each frame are parsed into K groups, where the i th group is assigned b_i bits.

Hence,

$$\sum_{i=1}^{K} b_i = B_f \ . \tag{8.7.3}$$

We may view the multicarrier modulator as generating K independent QAM subchannels, where the symbol rate for each subchannel is $\frac{1}{T}$ and the signal in each subchannel has a distinct QAM constellation.

Hence, the number of signal points in the constellation for the i th subchannel is $M_i = 2^{b_i}$.

Let X_k, $k = 0, 1, \cdots, K-1$, denote the complex-valued signal points corresponding to the information signals on the K subchannels.

These information symbols $\{X_k\}$ represent the values of the discrerte Fourier transform (DFT) of a multicarrier OFDM signal $x(t)$, where the modulation on each subcarrier is QAM.

Since $x(t)$ must be a real-valued signal, its N-point DFT $\{X_k\}$ must satisfy the symmetry property $X_{N-k} = X_k^*$.

Therefore, we create $N = 2K$ symbols from K information symbols by defining

$$X_{N-k} = X_k^*, \quad k = 1, 2, \cdots, K-1, \tag{8.7.4}$$

$$X_0^{'} = \mathrm{Re}(X_0) \tag{8.7.5}$$

$$X_N = \mathrm{Im}(X_0). \tag{8.7.6}$$

Note that the information symbol X_0 is split into two parts, both of which are real.

If we denote the new sequence of symbols as $\{X_k^{'},\ k = 0, 1, \cdots, N-1\}$, the N-point inverse DFT (IDFT) yields the real-valued sequence

$$x_n = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} X_k^{'} e^{j\frac{2\pi nk}{N}}, \qquad n = 0, 1, \cdots, N-1. \tag{8.7.7}$$

where $\frac{1}{\sqrt{N}}$ is simply a scale factor.

This sequence $\{x_n,\ 0 \le n \le N-1\}$ corresponds to samples of the multicarrier OFDM signal $x(t)$, consisting of K subcarriers, which may be expressed as

$$x(t) = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} X_k^{'} e^{j\frac{2\pi kt}{T}}, \qquad 0 \le t \le T, \tag{8.7.8}$$

where T is the signal duration and $x_n = x(\frac{nT}{N}), \quad n = 0, 1, \cdots, N-1.$

The signal samples $\{x_n\}$ generated by computing the IDFT are passed through a digital-to-analog (D/A) converter, where output, ideally is the OFDM signal waveform $x(t)$.

With $x(t)$ as the input to the channel, the channel output at the receiver may be expressed as

$$r(t) = x(t) * c(t) + n(t) \tag{8.7.9}$$

where $c(t)$ is the impulse response of the channel and $*$ stands for convolution.

Since the bandwidth Δf of each subchannel is selected to be very small relative to the overall channel bandwidth $W = K\Delta f$, the symbol duration $T = \dfrac{1}{\Delta f}$ is large compared to the duration of the channel impulse response.

To be specific, suppose that the channel impulse response spans $m+1$ signal samples, where $m << N$.

A simple way to completely avoid intersymbol interference (ISI) is to insert a time guard of duration $\dfrac{mT}{N}$ between transmission of successive data blocks.

This allows the response of the channel to die out before the next block of K symbols are transmitted.

An alternative method to avoid ISI is to append a so-called cyclic prefix to each block of N signal samples $\{x_n, \ \ 0 \le n \le N-1\}$.

The cyclic prefix for the block of samples consists of the samples $x_{N-m}, x_{N-m+1}, \cdots, x_{N-1}$.

These samples are appended to the beginning of the block, thus, creating a signal sequence of length $N+m$ samples, which may be indexed from $n=-m$ to $n=N-1$, where the first m samples constitute the cyclic prefix.

Then, if the sample values of the channel response are $\{c_n,\ 0 \le n \le m\}$, the convolution of $\{c_n\}$ with $\{x_n,\ -m \le n \le N-1\}$ produce the received signal $\{r_n\}$.

Since the ISI in any pair of successive signal transmission blocks affects the first m signal samples, we discard the first m samples of $\{r_n\}$ and demodulate the signal based on the received signal samples $\{r_n,\ 0 \le n \le N-1\}$.

If we view the channel characteristics in the frequency domain, the channel frequency response at the subcarrier frequencies $f_k = \dfrac{k}{T}$ is given by

$$C_k = C\left(\frac{2\pi k}{N}\right)$$

$$= \sum_{n=0}^{m} c_n e^{-j2\pi nk/N}, \qquad k = 0, 1, \cdots, N-1. \tag{8.7.10}$$

Since the ISI is eliminated by the use of either the cyclic prefix or the time quard band, the demodulated sequence of symbols may be expressed as

$$\hat{X}_k = C_k X_k^{'} + \eta_k, \quad k = 0, 1, \cdots, N-1, \tag{8.7.11}$$

where $\{\hat{X}_k\}$ is the output of the N-point DFT computed by the demodulator and

$\{\eta_k\}$ is the additive noise corrupting the signal.

As shown in Figure 8.50, the received signal is demodulated by computing the DFT of the received signal after it has been passed through an analog-to-digital (A/D) converter.

As in the case of the OFDM modulator, the DFT computation at the demodulator is performed efficiently by use of the FFT algorithm.

In order to recover the information symbols from the values of the computed DFT, it is necessary to estimate and compensate for the channel factors $\{C_k\}$.

The channel measurement can be accomplished by initially transmitting either a known modulated sequence on each of the subcarriers or, simply, transmitting the unmodulated subcarriers.

If the channel characteristics vary slowly with time, the time variations can be tracked by using the decisions at the output of the detector in a decision-directed manner.

Thus, the multicarrier OFDM system can be made to operate adaptively.

The transmission rate on each of the subcarriers can be optimized by properly allocating the average transmitted power and the number of bits that are transmitted by each subcarrier.

The SNR per subchannel may be defined as

$$\mathrm{SNR}_k = \frac{TP_k |C_k|^2}{\sigma_{nk}^2} \tag{8.7.12}$$

where T is the symbol duration,

P_k is the average transmitted power allocated to the k th subchannel,

$|C_k|^2$ is the squared magnitude of the frequency response of the k th subchannel, and

σ_{nk}^2 is the corresponding noise variance.

In subchannels with high SNR, we transmit more bits/symbol by using a larger QAM constellation compared to subchannels with low SNR.

Thus, the bit rate on each subchannel can be optimized in such a way that the error-rate performance among the subchannels is equalized to satisfy the desired specifications.

Multicarrier OFDM using QAM modulation on each of the subcarriers as described above has been implemented for a variety of applications, including high-speed transmission over telephone lines, such as digital subcarrier lines.

This type of multicarrier OFDM modulation has also been called **discrete-multitone** (DMT) **modulation**.

Multicarrier OFDM is also used in digital audio broadcasting in Europe and other parts of the world and in digital cellular communication systems.