Advanced Digitial Communications – WS 2001

Rüdiger Urbanke
EPFL
DSC-LTHC

January 30, 2002

Contents

1

REVIEW

1. HYPOTHESIS TESTING - DISCRETE CASE

Assume we have a set of *hypotheses* $H \in \{0, 1, \cdots, (m-1)\} := [m-1]$ with *priors* $p_i = \Pr\{H = i\}$ and that we observe the random variable $Y \in \mathbb{R}$, where we denote the *likelihood* of observing y under the hypothesis i by $f_{Y|H}(y|i)$.

Then, using Bayes' rule, the probability that i is the correct hypothesis given that y was observed, denoted by $p_{H|Y}(i|y)$, is equal to

$$p_{H|Y}(i|y) = \frac{f_{Y|H}(y|i)p_i}{f_Y(y)} = \frac{f_{Y|H}(y|i)p_i}{\sum_i f_{Y|H}(y|i)p_i}.$$

Note that for a fixed y we maximize the probability of correct decision if we choose the rule

$$\hat{H}(y) := \text{argmax}_i p_{H|Y}(i|y) = \text{argmax}_i f_{Y|H}(y|i)p_i.$$

This is true since, as observed above, $p_{H|Y}(i|y)$ *is* the probability that i is the correct hypothesis given that y was observed. This rule is called the *Maximum A Posteriori* (MAP) decision rule. Integrating over all observations y (weighted with $f_Y(y)$) we see that the MAP rule maximizes the probability of correct decision.

Of particular importance is the *Gaussian case*. Here we wish to distinguish between m given points $a_i \in \mathbb{R}^n$. Under hypothesis i, the observation Y is $Y = a_i + Z$, where $Z = (Z_1, \cdots, Z_n)$ is a jointly Gaussian vector of independent zero mean random variables each of variance σ^2. Therefore

$$f_{Y|H}(y|i) = \frac{1}{(2\pi\sigma^2)^{n/2}} e^{-\frac{\|y-a_i\|^2}{2\sigma^2}}.$$

Recall, that for any unit vector in $\mathbb{R}^n$ the projection of the vector Z onto this unit vector results in a zero mean Gaussian random variable with σ^2 and that the projection onto orthogonal dimensions are independent. In this case we see that the decision rule can be written as

$$\begin{aligned}
\hat{H}(y) &= \text{argmax}_i f_{Y|H}(y|i)p_i \\
&= \text{argmax}_i \frac{1}{(2\pi\sigma^2)^{n/2}} e^{-\frac{\|y-a_i\|^2}{2\sigma^2}} p_i \\
&= \text{argmax}_i -\frac{\|y-a_i\|^2}{2\sigma^2} + \ln(p_i) \\
&= \text{argmin}_i \frac{\|y-a_i\|^2}{2\sigma^2} - \ln(p_i) \qquad (1.1) \\
&= \text{argmax}_i -\frac{\|y\|^2}{2\sigma^2} + \frac{\langle y, a_i\rangle}{\sigma^2} - \frac{\|a_i\|^2}{2\sigma^2} + \ln(p_i) \\
&= \text{argmax}_i \frac{\langle y, a_i\rangle}{\sigma^2} - \frac{\|a_i\|^2}{2\sigma^2} + \ln(p_i). \qquad (1.2)
\end{aligned}$$

We see from (1.1) and (1.2) that in order to arrive at the optimal decision, we do not need to know the observation y itself but it suffices to know either the set of Euclidean distances $\{\|y-a_i\|^2\}_{i\in[m-1]}$ or the set of inner products $\{\langle y, a_i\rangle\}_{i\in[m-1]}$. We call such a quantity a *sufficient statistic*.

Under the assumption of uniform priors the decision rule has the following nice geometric interpretation depicted in Fig. 1.1: the decision regions equal the *Voronoi* regions of the set of points $\{a_0, \cdots, a_{m-1}\}$. More precisely, let Λ_i denote the decision region associated to hypothesis i, $i \in [m-1]$. We require that $\{\Lambda_i\}_{i\in[m-1]}$ partitions the whole space, i.e., the decision regions are disjoint and their union is equal to $\mathbb{R}^n$. We see from (1.1) that in the case of equal priors we have[1]

$$\begin{aligned}
\Lambda_i &= \{x \in \mathbb{R}^n : \|x-a_i\|^2 < \|x-a_j\|^2, \forall j \in [m-1]\setminus\{i\}\} \\
&= \{x \in \mathbb{R}^n : \langle x - \frac{a_i+a_j}{2}, a_i - a_j\rangle > 0, \forall j \in [m-1]\setminus\{i\}\},
\end{aligned}$$

where the first definition tells us that Λ_i is equal to the set of points which are closer (in terms of Euclidean distance) to hypothesis a_i than to any other hypothesis a_j, $j \neq i$, and where the second definition tells us that this region can be defined as the intersection of half spaces, each of them defined by means of a *hyperplane*. Note that in this equal prior case the decision rule is *independent* of σ^2, i.e., an optimal decision can be taken without knowing the magnitude of the noise variance.

In the general case with possibly nonuniform priors the decision regions are still given in terms of hyperplanes. To see this, it suffices to look at the case with

[1] In this definition we ignored boundary points, i.e., points which lie at equal (minimum) distance to several hypotheses. Those ties can be broken in an arbitrary manner without effecting the resulting probability of error.

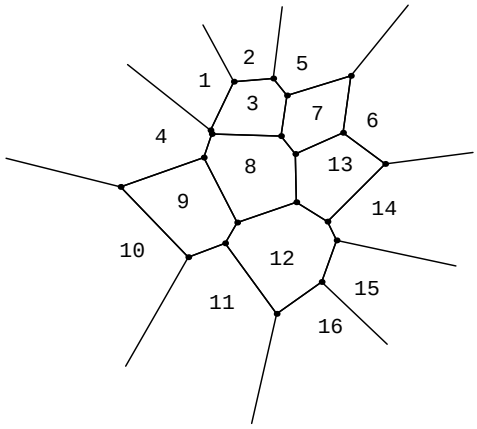


Figure 1.1: Example of optimal decision regions for the Gaussian case with uniform prior and sixteen hypotheses. The optimal decision regions are equal to the Voronoi regions.

only two hypotheses. In this case the decision rule can be written in the following form:

$$\begin{array}{lll} \frac{\langle y,a_0\rangle}{\sigma^2}-\frac{\|a_0\|^2}{2\sigma^2}+\ln(p_0) & \gtrless_{\hat{H}=1}^{\hat{H}=0} & \frac{\langle y,a_1\rangle}{\sigma^2}-\frac{\|a_1\|^2}{2\sigma^2}+\ln(p_1) \\ \frac{\langle y-\frac{a_0+a_1}{2},a_0-a_1\rangle}{\sigma^2} & \gtrless_{\hat{H}=1}^{\hat{H}=0} & \ln\frac{p_1}{p_0} \\ \langle y-\frac{a_0+a_1}{2},\frac{a_0-a_1}{\|a_0-a_1\|}\rangle & \gtrless_{\hat{H}=1}^{\hat{H}=0} & \frac{\sigma^2}{\|a_0-a_1\|}\ln\frac{p_1}{p_0} \end{array}$$

The geometric interpretation of this decision rule is shown in Fig. 1.2. Using this geometric interpretation we can immediately write down the respective error probabilities $\Pr\{e|H=0\}$ and $\Pr\{e|H=1\}$.[2] Denote by $Z_{\frac{a_0-a_1}{\|a_0-a_1\|}}$ the noise component in direction of the unit vector $\frac{a_0-a_1}{\|a_0-a_1\|}$. Note that by assumption $Z_{\frac{a_0-a_1}{\|a_0-a_1\|}}$ is

[2] Recall the definition of the $Q(\cdot)$ function,

$$Q(x) := \int_x^\infty \frac{1}{\sqrt{2\pi}} e^{-\frac{u^2}{2}}\, du.$$

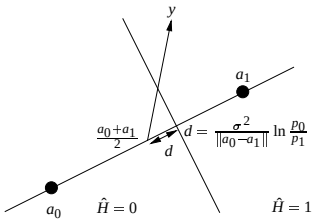


Figure 1.2: Geometric interpretation of the decision rule for the Gaussian case and $m = 2$.

Gaussian with variance σ^2. We therefore have

$$\begin{aligned}
\Pr\{e|H=0\} &= \Pr\{Z_{\frac{a_0-a_1}{\|a_0-a_1\|}} < -\frac{\|a_0-a_1\|}{2} - \frac{\sigma^2}{\|a_0-a_1\|}\} \\
&= Q\left(\frac{\|a_0-a_1\|}{2\sigma} + \frac{\sigma}{\|a_0-a_1\|}\ln\frac{p_0}{p_1}\right), \\
\Pr\{e|H=1\} &= Q\left(\frac{\|a_0-a_1\|}{2\sigma} - \frac{\sigma}{\|a_0-a_1\|}\ln\frac{p_0}{p_1}\right).
\end{aligned}$$

In terms of the Q-function we can express the probability of error for various popular modulation schemes.

Example 1. [QAM] In this case all four points are equivalent and we have the

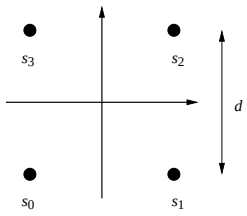


Figure 1.3: QAM constellation.

probability of correct decision, call it $\Pr\{c\}$,

$$\begin{aligned}
\Pr\{c\} &= \Pr\{c|H=0\} \\
&= \Pr\{Z_1 \le d/2\}\Pr\{Z_2 \le d/2\} \\
&= \left(1 - Q\left(\frac{d}{2\sigma}\right)\right)^2.
\end{aligned}$$

Therefore, the probability of error is equal to $\Pr\{e\} = 2Q\left(\frac{d}{2\sigma}\right) - Q\left(\frac{d}{2\sigma}\right)^2$.

□

2. IRRELEVANCE

In the sequel it will be handy to be able to refer to the following theorem.

Theorem 1. [Irrelevance] Let the hypothesis H take values in $\{0,1,\cdots,(m-1)\}$ and assume that the observation for a hypothesis testing problem is equal to $Y = (Y_1, Y_2)$. Then the MAP decision rule can be based on Y_1 alone if $f_{Y_2|Y_1,H}(y_2|y_1,i) = f_{Y_2|Y_1}(y_2|y_1)$. Conversly, this condition is also necessary for the decision metric to be independent of Y_2.

Proof. Using Bayes' rule and the assumption that $f_{Y_2|Y_1,H}(y_2|y_1,i) = f_{Y_2|Y_1}(y_2|y_1)$ we get

$$\begin{aligned}
\hat{H}(y_1,y_2) &:= \text{argmax}_i\Big(p_{H|Y_1,Y_2}(i|y_1,y_2)\Big) \\
&= \text{argmax}_i\Big(f_{Y_2|Y_1,H}(y_2|y_1,i)\frac{p_{H|Y_1}(i|y_1)}{f_{Y_2|Y_1}(y_2|y_1)}\Big) \\
&= \text{argmax}_i\Big(f_{Y_2|Y_1,H}(y_2|y_1,i)p_{H|Y_1}(i|y_1)\Big) \\
&= \text{argmax}_i\Big(p_{H|Y_1}(i|y_1)\Big),
\end{aligned}$$

which shows that in this case the MAP decision rule may be based on Y_1 alone.

Assume now that we require that the decision metric $f_{Y_2|Y_1,H}(y_2|y_1,i)p_{H|Y_1}(i|y_1)$ be independent of Y_2. Then we have

$$\begin{aligned}
& f_{Y_2|Y_1,H}(y_2|y_1,i)p_{H|Y_1}(i|y_1) = f_{Y_2,H|Y_1}(y_2,i|y_1) \\
= \quad & f_{Y_2|Y_1}(y_2|y_1)p_{H|Y_1,Y_2}(i|y_1,y_2) \stackrel{!}{=} f_{Y_2|Y_1}(y_2|y_1)p_{H|Y_1}(i|y_1),
\end{aligned}$$

which shows that $f_{Y_2|Y_1,H}(y_2|y_1,i) = f_{Y_2|Y_1}(y_2|y_1)$.

□

3. INNER PRODUCT SPACES AND GRAM-SCHMIDT ALGORITHM

In the previous section we have made some use of properties of the *inner product* of elements of $\mathbb{R}^n$. Several other inner product spaces will be important in the sequel and so we quickly review some basic facts about them. Let V be a vector space over the complex numbers.[3] An *inner product* on V is then a mapping $V^2 \to \mathbb{C}$, which we will denote by $\langle\cdot,\cdot\rangle$, such that for any triple $x,y,z \in V$ and any scalar $\alpha \in \mathbb{C}$,

- $\langle x+y,z\rangle = \langle x,z\rangle + \langle y,z\rangle$,

[3] In the same manner we can define an inner product over the real numbers.

- $\langle \alpha x, y \rangle = \alpha \langle x, y \rangle$,
- $\langle x, y \rangle = \langle y, x \rangle^*$,
- $\langle x, x \rangle > 0$, for $x \neq 0$.

Example 2. [$\mathbb{C}^n$] The *standard* inner product space is $\mathbb{C}^n$ with $\langle x, y \rangle = \sum_{i=1}^n x_i y_i^*$.

Example 3. Let V be the vector space of square integrable functions on $\mathbb{R}$, i.e., $V = L^2(\mathbb{R})$, either over $\mathbb{R}$ or over $\mathbb{C}$. It is easy to verify that

$$< x(t), y(t) > := \int_{\mathbb{R}} x(t) y^*(t)\, dt$$

specifies a well-defined inner product in this case.

Let V be an inner product space and let $A = \{a_1, \ldots, a_m\}$ be a set of elements of V. The following Gram-Schmidt procedure then allows us to find an *orthonormal basis* for A, let this basis be $\{\psi_1, \cdots, \psi_n\}$, $n \leq m$, so that

$$a_i = \sum_{j=1}^{n} \langle a_i, \psi_j \rangle \psi_j, \; i \in [n].$$

This basis is recursively defined by (ignoring cases of dependent vectors)

$$\begin{aligned}
\psi_1 &= \frac{a_1}{\sqrt{\langle a_1, a_1 \rangle}} \\
\psi_2 &= \frac{a_2 - \langle a_2, \psi_1 \rangle \psi_1}{\sqrt{a_2 - \langle a_2, \psi_1 \rangle \psi_1}} \\
\vdots &\quad \cdots \\
\psi_n &= \frac{a_m - \sum_{j=1}^{m-1} \langle a_m, \psi_j \rangle \psi_j}{\sqrt{a_m - \sum_{j=1}^{m-1} \langle a_m, \psi_j \rangle \psi_j}}.
\end{aligned}$$

4. HYPOTHESIS TESTING - CONTINUOUS CASE

Next consider a hypothesis testing scenario in which the hypotheses are elements of some set of signals $\{a_0(t), \cdots, a_{(m-1)}(t)\}$, where we assume that each a_i is square integrable. Assume further that the observation conditioned on the fact that the correct hypothesis is a_i is equal to

$$Y(t) = a_i(t) + Z(t),$$

where $Z(t)$ is a white Gaussian noise process with double-sided power spectral density $\frac{N_0}{2}$. Let $\{\psi_1(t), \cdots, \psi_n(t)\}$ be an orthonormal set for the space spanned by $\{a_0(t), \cdots, a_{(m-1)}(t)\}$. E.g., we can apply the Gram-Schmidt procedure to

$\{a_0(t),\cdots,a_{(m-1)}(t)\}$ to generate $\{\psi_1(t),\cdots,\psi_n(t)\}$. Let $a_i \in \mathbb{R}^n$ be the *expansion* of $a_i(t)$ with respect to $\{\psi_1(t),\cdots,\psi_n(t)\}$, i.e., $a_i = (a_{i1},\cdots,a_{in})$, where $a_{ij} = \langle a_i(t),\psi_j(t)\rangle$. Similarly, define $Z_j = \langle Z(t),\psi_j(t)\rangle$, so that $(Z_1,\cdots,Z_n)$. Then

$$Y(t) = \sum_{j=1}^{n} a_{ij}\psi_j(t) + \sum_{j=1}^{n} Z_j\psi_j(t) + Z^{\perp}(t).$$

This can be interpreted as follows: Think of the hypothesis as points in the n dimensional space. We now split the noise into that part which lives in this n-dimensional space and the part which is orthogonal to it. Since the orthogonal noise part is jointly independent of the transmitted hypothesis and the noise part within the n-dimensional space, by the Irrelevance Theorem 1 we can base our decision on the projection of $Y(t)$ onto the subspace spanned by $\{\psi_1(t),\cdots,\psi_n(t)\}$. Therefore, we are concerned with a Gaussian hypothesis testing problem where the set of hypotheses is equal to a set of m points $\{a_0,\cdots,a_{(m-1)}\}$ in $\mathbb{R}^n$. The observation Y in this case is $Y = a_i + Z$, where $Z = (Z_1,\cdots,Z_n)$ is a jointly Gaussian vector of independent zero mean random variables of variance $\sigma^2 = \frac{N_0}{2}$, see Exercise 1.1. But we have already seen how to solve this problem! Assume that we have uniform priors and look at the decision rule expressed in (1.2). This immediately gives rise to the *correlation* receiver shown in Fig. 1.4. Note that the

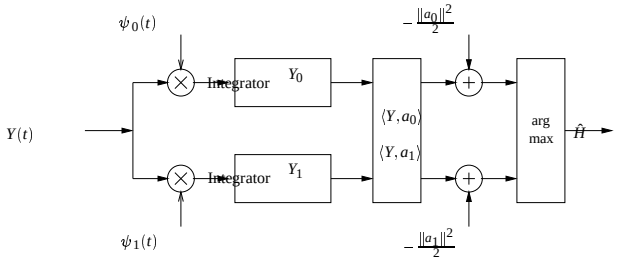


Figure 1.4: Correlation receiver.

important quantities $\langle Y(t),\psi_j(t)\rangle$ can be implemented either as *correlation* or by sampling the output of the filter $h(t) = \psi_j(-t)$ when input with the signal $Y(t)$ at time $t = 0$. The second approach is called a *matched filter*.

5. FOURIER AND Z-TRANSFORM

5.1 Z-TRANSFORM AND DISCRETE TIME FOURIER TRANSFORM

Assume we have a (complex valued) power series

$$\sum_{n\geq 0} a_n z^n$$

with the associated partial sums $A_n(z) = \sum_{m=0}^{n} a_n z^n$. Let the *radius of convergence* R be defined by

$$\frac{1}{R} = \limsup_{n\to\infty} |a|^{\frac{1}{n}}.$$

Recall from calculus that the series converges for $|x| < R$ and diverges for $|x| > R$. To see this, note that if $|x| = \alpha R$, $\alpha < 1$, then $|a_n x^n| \overset{\text{for } n \text{ suff. large}}{\leq} \left(\frac{|x|}{R}\right)^n \leq \alpha^n$ and that the series $\sum_{n\geq 0} \alpha^n$ converges for $\alpha < 1$. If on the other hand $|x| > R$ then by the definition of lim sup there exist infinitely many indixes k such that $|a_k x^k|^{\frac{1}{k}} \geq 1$ and therefore there exist infinitely many indices such that $|A_k(x) - A_{k-1}(x)| \geq 1$. It follows by the Cauchy criterion that $A_n(x)$ does not converge. It follows that if we are given a sum of the form

$$\sum_{n=-\infty}^{\infty} a_n z^n$$

then the region of convergence is an annular region (simply split the sum into the positive and the negative indices and observe that $x^{-k} = \left(\frac{1}{x}\right)^k$.)

Assume we have a discrete time (real or complex valued) signal x_n, $n \in \mathbb{Z}$. Its associated z-transform, call it $H(z)$ (if it exists), is *defined* by

Definition 1. [z-Transform]

$$H(z) \quad = \quad \sum_n h_n z^{-n}.$$

The inverse z-transform is usually accomplished by means of a *partial fraction* expansion, see Exercise 1.11.

Definition 2. [Basic Properties of z-Transform]

$$\begin{aligned} h^*_{-n} &\Leftrightarrow H^*(1/z^*) \\ h_{n-m} &\Leftrightarrow H(z)z^{-m} \\ \sum_k h_k g_{n-k} &\Leftrightarrow H(z)G(z) \\ \sum_k h_k g^*_{k-n} &\Leftrightarrow H(z)G^*(1/z^*) \end{aligned} \tag{1.3}$$

We say that a sequence h_n is *causal* if $h_n = 0$ for $n < 0$ and we say that it is *anticausal* if $h_n = 0$ for $n > 0$. For a causal sequence the ROC is of the form $|z| > R$ whereas for an anticausal it is of the form $|z| < R$. We say that a sequence is stable if $\sum_n |h_n| < \infty$. The ROC of a stable sequence must contain the *unit circle.* If $H(z)$, the z-Transform of h_n, is rational then this implies that for a stable and causal system all the poles of $H(z)$ must be within the unit circle. Finally, we say that a sequence h_n with rational z-Transform $H(z)$ is *minimum phase*, if all its poles and zeros are within the unit circle. Such a sequence has the property that

for all $N \geq 0$ it maximizes the quantity $\sum_{n=0}^{N} |h_n|^2$ over all sequences which have the same $|H(z)|$.

The Discrete-Time Fourier Transform (DTFT) of h_n is defined as

Definition 3. [DTFT]

$$\begin{aligned} H(e^{2\pi jf}) &= \sum_n h_n e^{-2\pi jfn}, \\ h_n &= \int_0^1 H(e^{2\pi jf}) e^{2\pi jfn} df. \end{aligned}$$

Its basic properties follow immediately from the one of the z-Transform if we observe that

$$H(e^{2\pi jf}) = H(z)|_{z=e^{2\pi jf}},$$

(assuming that the ROC of $H(z)$ contains the unit circle), hence the notation $H(e^{2\pi jf})$.

Definition 4. [Fourier Transform]

$$\begin{aligned} H(f) &= \int_{-\infty}^{\infty} h(t) e^{-2\pi jft} dt \\ h(t) &= \int_{-\infty}^{\infty} H(f) e^{2\pi jft} df \end{aligned}$$

5.2 FOURIER TRANSFORM

Definition 5. [Properties of Fourier Transform]

$$h^*(-t) \Leftrightarrow H^*(f) \tag{1.4}$$

$$h(t-s) \Leftrightarrow H(f) e^{-2\pi jsf} \tag{1.5}$$

$$h(t/a) \Leftrightarrow aH(fa) \tag{1.6}$$

$$\text{sinc}(t) = \frac{\sin(\pi t)}{\pi t} \Leftrightarrow \text{rect}(f) = \begin{cases} 1, & |f| \leq \frac{1}{2}, \\ 0, & |f| > \frac{1}{2}. \end{cases} \tag{1.7}$$

$$\int_{-\infty}^{\infty} h(\tau) g(t-\tau) d\tau \Leftrightarrow H(f) G(f) \tag{1.8}$$

$$\int_{-\infty}^{\infty} h(\tau) g^*(\tau - t) d\tau \Leftrightarrow H(f) G^*(f) \tag{1.9}$$

$$\int_{-\infty}^{\infty} h(t) g^*(t) dt = \int_{-\infty}^{\infty} H(f) G^*(f) df \tag{1.10}$$

Example 4. [Basic Properties of sinc Function] Using the above relations we get.

$$\begin{aligned}
\operatorname{sinc}(t) = \frac{\sin(\pi t)}{\pi t} \quad &\Leftrightarrow \quad \operatorname{rect}(f) := \begin{cases} 1, & |f| \le \frac{1}{2}, \\ 0, & |f| > \frac{1}{2}. \end{cases} \\
\operatorname{sinc}\left(\frac{t}{\tau}\right) \quad &\Leftrightarrow \quad \begin{cases} \tau, & |f| \le \frac{1}{2\tau}, \\ 0, & |f| > \frac{1}{2\tau}. \end{cases} \\
\operatorname{sinc}\left(\frac{t}{\tau} - n\right) = \operatorname{sinc}\left(\frac{t - \tau n}{\tau}\right) \quad &\Leftrightarrow \quad \begin{cases} \tau e^{-2\pi j n \tau f}, & |f| \le \frac{1}{2\tau}, \\ 0, & |f| > \frac{1}{2\tau}. \end{cases} \\
\int_{-\infty}^{\infty} \operatorname{sinc}\left(\frac{t}{\tau} - n\right) \operatorname{sinc}\left(\frac{t}{\tau} - m\right) dt \quad &= \quad \int_{-\frac{1}{2\tau}}^{\frac{1}{2\tau}} \tau^2 e^{-2\pi j (n-m)\tau f} df = \begin{cases} 0, & m \neq n, \\ \tau, & n = m. \end{cases}
\end{aligned}$$

From the last equality we conclude that $\operatorname{sinc}\left(\frac{t}{\tau}\right)$ is orthogonal to all of its shifts (by multiplies of τ)! Further, we see that the functions $\sqrt{\frac{1}{\tau}}\operatorname{sinc}\left(\frac{t}{\tau} - n\right)$, $n \in \mathbb{Z}$, form an orthonormal set. One can also show that this set is complete for the class of square integrable functions which are low-pass limited to $\frac{1}{2\tau}$. □

6. SAMPLING THEOREM

Theorem 2. [Sampling Theorem] Let $f(t)$ be a square integrable function which is low-pass limited to W. Then $f(t)$ is specified by its values at a sequence of points spaced $\tau = \frac{1}{2W}$ apart. In particular,

$$f(t) = \sum_{-\infty}^{\infty} f(n\tau) \operatorname{sinc}\left(\frac{t}{\tau} - n\right).$$

7. NYQUIST CRITERION

Theorem 3. [Nyquist Criterion] A function $\psi(t)$ and its shifts $\psi(t - n\tau)$ form an orthonormal set if and only if

$$\sum_{-\infty}^{\infty} |\Psi(f - \frac{k}{\tau})|^2 = \tau, \quad -\frac{1}{2\tau} \le f \le \frac{1}{2\tau}.$$

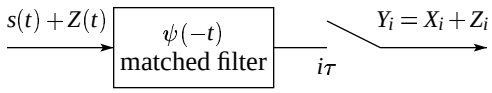


Figure 1.5: Optimal receiver structure.

Proof.

$$\begin{cases} 1, & n=0, \\ 0, & n\neq 0 \end{cases} \overset{\text{assumption}}{=} \int_{-\infty}^{\infty} \psi(t-\tau n)\psi^*(t)\,dt$$
$$\overset{(1.4)\ \&\ (1.5)}{=} \int_{-\infty}^{\infty} |\Psi(f)|^2 e^{-2\pi j n\tau f}\,df$$
$$\overset{\tau f\to f}{=} \frac{1}{\tau}\int_{\frac{1}{2}}^{-\frac{1}{2}} \left(\sum_k |\Psi(\frac{f-k}{\tau})|^2\right) e^{-2\pi j n f}\,df.$$

Note that the right hand side is equal to $\frac{1}{\tau}$ times the n-th Fourier coefficient of the function $\left(\sum_k |\Psi(\frac{f-k}{\tau})|^2\right)$. From the left hand side we see that this function only has a DC term and that this DC term is equal to one. From this the claim follows. □

8. TRANSMISSION OVER THE BANDLIMITED AWGN CHANNEL

Assume that we use a Nyquist pulse $\psi(t)$ to generate the signal

$$s(t) = \sum_{i=-\infty}^{\infty} X_i \psi(t-i\tau).$$

Assume further that $\Psi(f)$ is band limited to W, $W \geq \frac{1}{2\tau}$, and that we use $s(t)$ to transmit over a band-limited AWGN channel with bandwidth at least W. Then the optimal receiver structure is as shown in Fig. 1.5.

I.e., the continuous waveform channel is converted into the equivalent discrete time channel $Y_i = X_i + Z_i$, where Z_i is a sequence of i.i.d. zero mean Gaussian random variables with variance $\sigma^2 = \frac{N_0}{2}$.

9. COMPLEX GAUSSIAN RANDOM VARIABLES AND PROCESSES

We have already seen that if $Z = (Z_1, \ldots, Z_n)$ denotes a real valued zero mean jointly Gaussian vector with independent components each of variance σ^2 then

$$f_Z(z) = \frac{1}{(2\pi\sigma^2)^{n/2}} e^{-\frac{\|z\|^2}{2\sigma^2}}.$$

In the general (correlated) case the density function $f_Z(z)$ has the form

$$f_Z(z) = \frac{1}{(2\pi)^{n/2}\sqrt{|K|}} e^{-\frac{1}{2}(z-\mu)K^{-1}(z-\mu)^T},$$

where mean μ and covariance K are defined by $\mu = \mathbb{E}[Z]$ and $K = \mathbb{E}[(Z-\mu)^T(Z-\mu)]$. Note that in the Gaussian case mean and covariance completely specify the probability density function and that uncorrelatedness implies independence.

Consider now a *complex-valued* Gaussian random variable Z, i.e.,

$$Z = R + jI,$$

where R and I denote the real and imaginary components and where we assume that (R,I) is jointly Gaussian. Assume that Z has zero mean. It follows that the probability density function of Z is completely specified by the second order statistics of (R,I), i.e., by

$$K = \begin{pmatrix} \mathbb{E}[R^2] & \mathbb{E}[RI] \\ \mathbb{E}[RI] & \mathbb{E}[I^2] \end{pmatrix}.$$

Let

$$\mathcal{R}_Z = \mathbb{E}[ZZ^*] = \mathbb{E}[R^2] + \mathbb{E}[I^2] \tag{1.11}$$

denote the correlation. As we can see, in the complex case the correlation is not sufficient to specify the pdf of the random variable. Define the *complementary correlation* to be

$$\tilde{\mathcal{R}}_Z = \mathbb{E}[ZZ] = \mathbb{E}[R^2] - \mathbb{E}[I^2] + 2j\mathbb{E}[RI]. \tag{1.12}$$

We can see that both $\mathcal{R}_Z$ and $\tilde{\mathcal{R}}_Z$ are completely specified by the triple $\mathcal{R}_R$, $\mathcal{R}_I$ and $\mathcal{R}_{RI}$ and vice versa

$$\begin{aligned} \mathcal{R}_R &= \frac{\mathrm{Re}\left\{\mathcal{R}_Z + \tilde{\mathcal{R}}_Z\right\}}{2}, \\ \mathcal{R}_I &= \frac{\mathrm{Re}\left\{\mathcal{R}_Z - \tilde{\mathcal{R}}_Z\right\}}{2}, \\ \mathcal{R}_{RI} &= \frac{\mathrm{Im}\left\{-\mathcal{R}_Z + \tilde{\mathcal{R}}_Z\right\}}{2} = \frac{\mathrm{Im}\left\{\tilde{\mathcal{R}}_Z\right\}}{2}. \end{aligned}$$

Further, we see that the real part and the complex part of Z are equal variance and uncorrelated (and hence independent) if and only if $\tilde{\mathcal{R}}_Z = 0$ and that in this case they also have the same variance. Therefore, we will call a complex-valued zero mean Gaussian random variable *circularly symmetric* iff $\tilde{\mathcal{R}}_Z = 0$. The variance in each component is then simply half the total variance, i.e., $\mathcal{R}_R = \mathcal{R}_I = \frac{1}{2}\mathcal{R}_Z$.

We will now generalize to complex-valued Gaussian vectors. Let $Z = (Z_1, \ldots, Z_n)$ be a vector of zero-mean complex-valued Gaussian random variables. If the distribution of any subset of real and imaginary components is jointly Gaussian we

will say that Z is a complex-valued Gaussian vector. Again we will assume that Z has zero mean and, as in the real case, this implies that the pdf of Z is completely specified by its second order statistics. It is easy to see that, similar to the one dimensional case, the quantities $\mathcal{R}_{R_iR_j}$, $\mathcal{R}_{I_iI_j}$ and $\mathcal{R}_{R_iI_j}$ specify the correlation

$$\begin{aligned}\mathcal{R}_{Z_iZ_j} &= \mathbb{E}[Z_i Z_j^*] \\ &= \mathbb{E}[R_iR_j] + \mathbb{E}[I_iI_j] - j\mathbb{E}[R_iI_j] + j\mathbb{E}[R_jI_i] \\ &= \mathcal{R}_{R_iR_j} + \mathcal{R}_{I_iI_j} - j\mathcal{R}_{R_iI_j} + j\mathcal{R}_{R_jI_i},\end{aligned}$$

and the complementary correlation

$$\begin{aligned}\tilde{\mathcal{R}}_{Z_iZ_j} &= \mathbb{E}[Z_i Z_j] \\ &= \mathbb{E}[R_iR_j] - \mathbb{E}[I_iI_j] + j\mathbb{E}[R_iI_j] + j\mathbb{E}[R_jI_i] \\ &= \mathcal{R}_{R_iR_j} - \mathcal{R}_{I_iI_j} + j\mathcal{R}_{R_iI_j} + j\mathcal{R}_{R_jI_i}.\end{aligned}$$

Vice versa, we have

$$\begin{aligned}\mathcal{R}_{R_iR_j} &= \frac{\mathrm{Re}\left\{\mathcal{R}_{Z_iZ_j} + \tilde{\mathcal{R}}_{Z_iZ_j}\right\}}{2}, \\ \mathcal{R}_{I_iI_j} &= \frac{\mathrm{Re}\left\{\mathcal{R}_{Z_iZ_j} - \tilde{\mathcal{R}}_{Z_iZ_j}\right\}}{2}, \\ \mathcal{R}_{R_iI_j} &= \frac{\mathrm{Im}\left\{-\mathcal{R}_{Z_iZ_j} + \tilde{\mathcal{R}}_{Z_iZ_j}\right\}}{2}, \\ \mathcal{R}_{R_jI_i} &= \frac{\mathrm{Im}\left\{\mathcal{R}_{Z_iZ_j} + \tilde{\mathcal{R}}_{Z_iZ_j}\right\}}{2}.\end{aligned}$$

Assume now that for all i and j we have $\tilde{\mathcal{R}}_{Z_iZ_j} = 0$ and that for $i \neq j$, $\mathcal{R}_{Z_iZ_j} = 0$. In this case we conclude that any subset of real and imaginary components are uncorrelated and, hence, independent and that all real and imaginary components have the same variance. We will again say in this case that the random variables are circularly symmetric. Note that such a process is completely specified by its power spectral density.

We can translate this concept to complex-valued Gaussian processes. A complex-valued zero-mean Gaussian process is circularly symmetric if

$$\mathbb{E}[Z(t)Z(t-\tau)] = 0, \ \ \forall t, \tau.$$

Note that such a process is strict sense stationary if and only if it is wide sense stationary.

Next note that the property of being circularly symmetric is preserved by linear time-invariant filtering, since if $Z(t)$ is circularly symmetric and if $Y(t)$ is the result of passing $X(t)$ through a linear time-invariant filter with impulse response $h(t)$ then

$$\mathbb{E}[Y(t)Y(t-\tau)] = \int\int h(\alpha)h(\beta)\mathbb{E}[X(t-\alpha)X(t-\tau-\beta)]\, d\alpha\, d\beta = 0.$$

10. FILTERING OF WIDE SENSE STATIONARY STOCHASTIC PROCESSES

Let $X(t)$ denote a (real or complex valued) wide sense stationary stochastic process with mean $m_X = \mathbb{E}[X(t)]$ and autocorrelation function $\mathcal{R}_X(\tau)$,

$$\mathcal{R}_X(\tau) = \mathbb{E}[X(t)X^*(t-\tau)].$$

Recall that the *power* of the stochastic process is equal to

$$\mathcal{R}_X(0) = \mathbb{E}[|X(t)|^2].$$

Further, the *power spectral density* associated to $X(t)$, denoted by $\mathcal{S}_X(f)$, is equal to the Fourier transform of $\mathcal{R}_X(\tau)$, i.e.,

$$\mathcal{S}_X(f) = \int \mathcal{R}_X(\tau) e^{-2\pi j \tau f}\, df.$$

Note that $\mathcal{R}_X(\tau)$ is conjugate symmetric, i.e, $\mathcal{R}_X(\tau) = \mathcal{R}_X^*(-\tau)$, so that $\mathcal{S}_X(f)$ is real valued.

Let $Y(t)$ denote the result of passing $X(t)$ through a linear time invariant filter with impulse response $h(t)$, i.e.,

$$Y(t) = \int X(\tau) h(t-\tau)\, d\tau.$$

We then have

$$\begin{aligned}
\mathcal{R}_Y(\tau) &= \mathbb{E}[Y(t)Y^*(t-\tau)] \\
&= \mathbb{E}\left[\left(\int h(\alpha) X(t-\alpha)\, d\alpha \right) \left(\int h^*(\beta) X^*(t-\tau-\beta)\, d\beta \right) \right] \\
&= \int_\alpha h(\alpha) \left(\int_\beta h^*(\beta) \mathbb{E}[X(t-\alpha)X^*(t-\tau-\beta)]\, d\beta \right) d\alpha \\
&= \int_\alpha h(\alpha) \left(\int_\beta h^*(\beta) \mathcal{R}_X(\tau-\alpha+\beta)\, d\beta \right) d\alpha \\
&= \int_\alpha h(\alpha) \left(\int_\beta h^*(-\beta) \mathcal{R}_X((\tau-\alpha)-\beta)\, d\beta \right) d\alpha \\
&= (h(\tau) * (h^*(-\tau) * \mathcal{R}_X(\tau))).
\end{aligned}$$

In the frequency domain this translates to

$$\mathcal{S}_Y(f) = \mathcal{S}_X(f)|H(f)|^2.$$

From this relationship we see that $\mathcal{S}_X(f)$ must also be non-negative since if we let $H(f)$ be a narrow bandpass filter centered around some frequency f_0 then we see that the power of the process $X(t)$ "which is located around frequency f_0" is proportional to $\mathcal{S}_X(f_0)$.

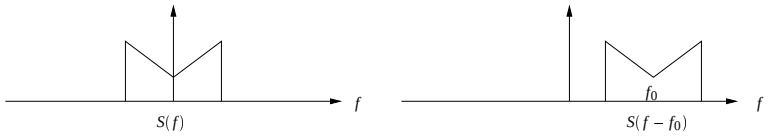


Figure 1.6: Modulation of a signal.

11. PASSBAND SYSTEMS

Assume we have a real valued signal $s(t)$ and its Fourier transform $S(f)$. Then we have the conjugacy constraint $S(f) = S^*(-f)$. Further we know from the *modulation property* that

$$s(t)e^{2\pi j f_0 t} \Leftrightarrow S(f - f_0).$$

This relationship is shown in Fig. 1.6. Define $H_>(f)$ as

$$H_>(f) := \begin{cases} 1, & f > 0, \\ \frac{1}{2}, & f = 0, \\ 0, & f < 0. \end{cases}$$

If $s(t)$ is an arbitrary real-valued signal, we *define* $\hat{s}(t)$ to be the signal with Fourier transform

$$\hat{S}(f) := \sqrt{2}S(f)H_>(f).$$

The factor $\sqrt{2}$ ensures that $s(t)$ and $\hat{s}(t)$ have *equal energy*. A signal $\hat{s}(t)$ such that $\hat{S}(f) = 0$ for $f < 0$ is called *analytic*. In the time domain this relationship can be expressed as

$$\hat{s}(t) = s(t) * \sqrt{2}h_>(t),$$

where $h_>(t) \Leftrightarrow H_>(f)$ constitutes a Fourier transform pair. We claim that the inverse relationship which permits us to go from $\hat{s}(t)$ back to $s(t)$ is given by

$$s(t) = \sqrt{2}\,\mathrm{Re}\{\hat{s}(t)\}.$$

This is most easily seen from

$$\begin{aligned} \sqrt{2}\,\mathrm{Re}\{\hat{s}(t)\} &= \sqrt{2}\frac{(\hat{s}(t) + \hat{s}^*(t))}{2} \\ &= \frac{\hat{s}(t)}{\sqrt{2}} + \frac{\hat{s}^*(t)}{\sqrt{2}} \\ &\Leftrightarrow \underbrace{\frac{\hat{S}(f)}{\sqrt{2}}}_{S(f) \text{ for } f>0} + \underbrace{\frac{\hat{S}^*(-f)}{\sqrt{2}}}_{S(f) \text{ for } f<0} \\ &= S(f). \end{aligned}$$

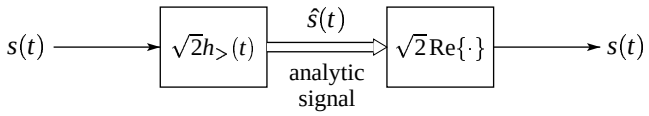


Figure 1.7: Relationship between $s(t)$ and $\hat{s}(t)$.

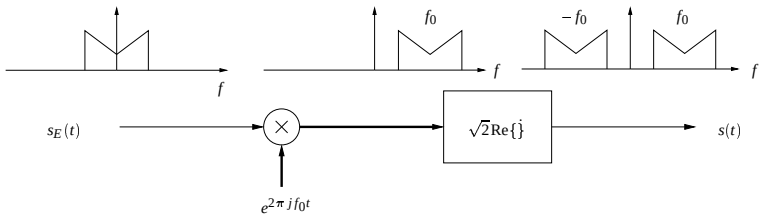


Figure 1.8: Up-conversion of a signal. The output signal $s(t)$ is equal to $s(t) = \sqrt{2}\mathrm{Re}\{s_E(t)e^{2\pi j f_0 t}\} = \sqrt{2}s_E(t)\cos(f_0 t)$.

We see that $\sqrt{2}\mathrm{Re}\{\hat{s}(t)\}$ and $s(t)$ have the same Fourier transform and are hence identical. We summarize this observation in Fig. 1.7.

Using the above notation, the up/down-conversion of a signal can be described in a compact form. Let $s_E(t)$ be a baseband signal of bandwidth W. By this we mean that

$$S_E(f) = 0, \ |f| > W.$$

The *up-conversion* is shown in Fig. 1.8. We multiply the signal $s_E(t)$ by $e^{2\pi j f_0 t}$, scale it by $\sqrt{2}$ and take the real part. We can recover our original signal by performing a *down-conversion*:

$$s(t) \to \hat{s}(t) \to \hat{s}(t)e^{-2\pi j f_0 t} = s_E(t).$$

Note that $s_E(t)$ has bandwidth W whereas the up-converted signal occupies a bandwidth of $2W$. It appears that we loose a factor two in bandwidth efficiency. But we can gain back this factor two by up-converting *two* real-valued baseband signals of bandwidth W into a *single* bandpass signal of bandwidth $2W$. Let $s_I(t)$ and $s_Q(t)$ denote two real valued baseband signals of bandwidth W and define

$$s_E(t) := s_I(t) + j s_Q(t).$$

Let

$$s(t) := \sqrt{2}\mathrm{Re}\{s_E(t)e^{2\pi j f_0 t}\} = \sqrt{2}s_I(t)\cos(2\pi f_0 t) - \sqrt{2}s_Q(t)\sin(2\pi f_0 t)$$

be the up-converted signal.[4] We can then recover our original two signals by performing the standard down-conversion,

$$s(t) \to \hat{s}(t) \to \hat{s}(t)e^{-2\pi j f_0 t} = s_E(t).$$

Why is it useful to be able to translate a signal into different frequency ranges? In this way we can perform all signal processing tasks at a convenient frequency range (maybe baseband) and we can choose this frequency range independent of the actual transmission band. In this way only a small part of the transmitter/receiver architecture depends on the actual transmission frequency.

So far we have started with a (complex valued) baseband signal of bandwidth W and up-converted it into a real-valued passband signal of bandwidth $2W$. But we can also reverse this process. Assume we have given a real-valued passband signal $s(t)$ of bandwidth W. Then we can define a baseband *equivalent* signal $s_E(t)$ (hence the "E"), which in general is complex-valued and has bandwidth $W/2$.

Below we summarize the relationships between the real-valued signal $s(t)$, its corresponding analytic signal $\hat{s}(t)$ and the baseband equivalent signal $s_E(t)$:

$$\begin{array}{ccccc}
 & \xrightarrow{*\sqrt{2}h_>(t)} & & \xrightarrow{e^{-2\pi j f_0 t}} & \\
s(t) & & \hat{s}(t) & & s_E(t) \\
 & \xleftarrow{\sqrt{2}\mathrm{Re}\{\cdot\}} & & \xleftarrow{e^{2\pi j f_0 t}} & \\
\Updownarrow \mathrm{FT} & & \Updownarrow \mathrm{FT} & & \Updownarrow \mathrm{FT} \\
 & \xrightarrow{\sqrt{2}H_>(f)} & & \xrightarrow{f \to f+f_0} & \\
S(f) & & \hat{S}(f) & & S_E(f) \\
 & \xleftarrow{} & & \xleftarrow{f \to f-f_0} &
\end{array}$$

Lets assume that we transmit the passband signal $s(t)$ through a passband channel with impulse response $h(t)$. Let us denote the channel output by $w(t)$, i.e.,

$$w(t) = s(t) * h(t) \Leftrightarrow W(f) = S(f)H(f).$$

Using above relationships we conclude that

$$\begin{aligned}
w(t) = s(t) * h(t) \quad &\Longleftrightarrow \quad W(f) = S(f)H(f) \\
\hat{w}(t) = \hat{s}(t) * \frac{1}{\sqrt{2}}\hat{h}(t) \quad &\Longleftrightarrow \quad \hat{W}(f) = S_E(f-f_0)\frac{1}{\sqrt{2}}H_E(f-f_0) \\
w_E(t) = s_E(t) * \frac{1}{\sqrt{2}}h_E(t) \quad &\Longleftrightarrow \quad W_E(f) = S_E(f)\frac{1}{\sqrt{2}}H_E(f). \qquad (1.13)
\end{aligned}$$

[4] Recall the following relationships:

$$\cos(x) = \frac{e^{jx}+e^{-jx}}{2}, \quad \sin(x) = \frac{e^{jx}-e^{-jx}}{2j}, \quad e^{jx} = \cos(x) + j\sin(x), \quad e^{-jx} = \cos(x) - j\sin(x)$$

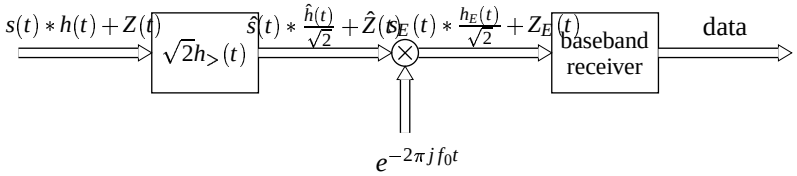


Figure 1.9: Receiver for passband signal $s(t)$. Note that $Z_E(t)$ is complex valued $Z_E(t) = Z_I(t) + jZ_Q(t)$, where the two components are independent and each have a two-sided power spectral density equal to $\frac{N_0}{2}$.

Now lets see how we can find a baseband equivalent form for passband Gaussian noise. Assume that $Z(t)$ has power spectral density equal to

$$\mathcal{S}_Z := \begin{cases} \frac{N_0}{2}, & f_0 - W < |f| < f_0 + W, \\ 0, & \text{elsewhere.} \end{cases}$$

Since $\hat{Z}(t)$ is the result of passing $Z(t)$ through a time-invariant filter we conclude that $\hat{Z}(t)$ is a zero-mean WSS stochastic process with power spectral density equal to

$$\mathcal{S}_{\hat{Z}} := \mathcal{S}_Z |\sqrt{2}H_>(f)|^2 = \begin{cases} N_0, & f_0 - W < f < f_0 + W, \\ 0, & \text{elsewhere.} \end{cases}$$

Now let $Z_E(t) = \hat{Z}(t)e^{-2\pi j f_0 t}$. Then the autocorrelation of $Z_E(t)$ is equal to

$$\begin{aligned} \mathcal{R}_{Z_E}(t,s) &= \mathbb{E}\left[\hat{Z}(t)e^{-2\pi j f_0 t}\hat{Z}^*(s)e^{2\pi j f_0 s}\right] \\ &= \mathbb{E}\left[\hat{Z}(t)\hat{Z}^*(s)\right]e^{-2\pi j f_0 (t-s)} \\ &= \mathcal{R}_{\hat{Z}}(t-s)e^{-2\pi j f_0 (t-s)}. \end{aligned}$$

We conclude that $Z_E(t)$ is also WSS with power spectral density equal to

$$\mathcal{S}_{Z_E}(f) = \mathcal{S}_{\hat{Z}}(f + f_0) = \begin{cases} N_0, & -W < f < W, \\ 0, & \text{elsewhere.} \end{cases}$$

Further, one can show that $Z_I(t)$ and $Z_Q(t)$, the real and imaginary parts of $Z_E(t)$, are uncorrelated and have a power spectral density of

$$\mathcal{S}_{Z_I}(f) = \mathcal{S}_{Z_Q}(f) = \begin{cases} \frac{N_0}{2}, & -W < f < W, \\ 0, & \text{elsewhere.} \end{cases}$$

Let $s_E(t)$ be the (complex) baseband signal and $s(t)$ the corresponding passband signal. The receiver for $s(t)$ looks as in Fig. 1.9. The channel seen from the up-converter input to the down-converted output is as shown in Fig. 1.10.

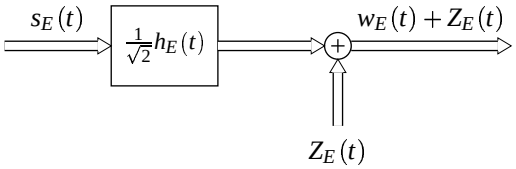


Figure 1.10: Channel seen from the up-converter input to the down-converted output. All quantities are complex valued.

12. JUST ENOUGH ABOUT FORMAL POWER SUMS

In case you did not feel comfortable about the manipulations above (in particular about the division in the very last step), here are some explanations:

Given two formal power sums $x(D) := \sum_{n\geq 0} x_n D^n$ and $y(D) := \sum_{n\geq 0} y_n D^n$ we can *define* their *addition* in the following natural way

$$x(D) + y(D) := \sum_{n\geq 0} (x_n + y_n) D^n.$$

In a similar way we can *define* their *multiplication* by

$$x(D) \cdot y(D) := \sum_{n\geq 0} \left(\sum_{i=0}^{n} x_i y_{n-i} \right) D^n,$$

which is the rule familiar from polynomial multiplication. Note that this is well defined, since in order to compute the n-th coefficient of the product we only need to perform a *finite* number of operations.

We next look if it is possible to define *division*. Recall that over the reals we say that $y = \frac{1}{x}$, $x \neq 0$, if $z := xy = 1$, and we say that y is the *multiplicative inverse* of x. Dividing by x is then the same as multiplying by y. We will proceed along the same lines for formal power sums. Consider the formal power sum $x(D)$. We want to find the formal power sum $y(D)$ such that $z(D) := x(D)y(D) = 1$. We will then say that $y(D)$ is the multiplicative inverse of $x(D)$ and we can then divide by $x(D)$ by multiplying with $y(D)$. Using the multiplication rule from above, we get the following set of equations:

$$\begin{aligned} 1 &= z_0 = x_0 y_0, \\ 0 &= z_i = \sum_{j=0}^{i} y_j x_{i-j}, \quad i \geq 1. \end{aligned}$$

We see that this set of equations has a solution (and that this solution is unique) if

and only if $x_0 \neq 0$. In this case we get

$$\begin{aligned} y_0 &= \frac{1}{x_0}, \\ y_i &= \frac{1}{x_0}\sum_{j=0}^{i-1} y_j x_{i-j}, \quad i \geq 1. \end{aligned}$$

Since again the evaluation of each coefficient y_i only involves a finite number of algebraic operations and only makes use of the values of y_j, $j < i$, this gives rise to a well-defined formal power sums. In summary, a formal power sums $x(D)$ has a multiplicative inverse iff $x_0 \neq 0$. The above procedure of finding $\frac{1}{x(D)}$ is also called *long division*.

Example 5. [Inverse of $1+D$] Consider the example $x(D) = 1+D$. Since $x_0 \neq 0$, $\frac{1}{1+D}$ exists. We get $y_0 = \frac{1}{x_0} = 1$, $y_1 = \frac{1}{x_0} y_0 x_1 = 1$, $y_2 = \frac{1}{x_0}(y_0 x_2 + y_1 x_1) = y_1 = 1$, and, in general, $y_i = \frac{1}{x_0}\sum_{j=0}^{i-1} y_j x_{i-j} = \frac{1}{x_0} y_{i-1} x_1 = y_{i-1} = 1$. Therefore $y(D) = \sum_{n=0}^{\infty} D^i$.[5] □

HISTORICAL NOTES

EXERCISES

1.1 (Transformation of Gaussian Random Variables). Let $Z = (Z_1, \ldots, Z_n)$ denote a jointly Gaussian vector with independent components with zero mean and each with variance σ^2, i.e., we have

$$f_Z(z) = \frac{1}{(2\pi\sigma^2)^{n/2}} e^{-\frac{\|z\|^2}{2\sigma^2}}$$

Let $\{\psi_1, \ldots, \psi_n\}$ be any basis for $\mathbb{R}^n$, i.e., an orthonormal set and let $W = (W_1, \ldots, W_n)$ denote a random vector whose components are the projections of Z onto this basis, i.e, $W_i = \langle Z, \psi_i \rangle$. Show that W has the same distribution as Z, i.e., W is a jointly Gaussian vector with independent components with zero mean and each with variance σ^2.

1.2 (Convexity of Voronoi Regions). Recall that for the Gaussian hypothesis testing case with uniform priors the decision regions are equal to the Voronoi regions. Prove that in this case the decision regions are convex regions, i.e., if x_1, x_2 are elements of the decision region associated to hypothesis i then so is $\alpha x_1 + (1-\alpha)x_2$, where $\alpha \in [0,1]$.

[5] The resemblance to the summation formula $\sum_{i=0}^{\infty} x^i = \frac{1}{1-x}$, where $|x| < 1$, is not a coincidence. In general, as a rule of thumb any identity which is valid for Taylor series and which can be meaningfully interpreted in the realm of formal power series will still be valid if considered as an identity of formal power series.

1.3 (8-PSK Constellation). Consider the Gaussian hypothesis testing problem with $H \in \{a_0, \cdots, a_7\}$, $a_i = e^{\frac{2\pi j}{8} i}$, $i = 0, \cdots, 7$, i.e., the hypotheses form an 8-PSK constellation and the observation assuming that the correct hypothesis is i is equal to $Y = a_i + Z$, $Z = (Z_1, Z_2)$, $Z_k \sim \mathcal{N}(0, \sigma^2)$ and Z_1 is independent of Z_2. What is the *exact* probability of error of the MAP decision rule and what does the union bound give?

1.4. Consider again a Gaussian hypothesis testing problem with $m = 2$. Under hypothesis $H = 0$ the transmitted point is equally likely to be $a_{00} = (1,1)$ or $a_{01} = (-1,-1)$, whereas under hypothesis $H = 1$ the transmitted point is equally likely to be $a_{10} = (-1,1)$ or $a_{11} = (1,-1)$. Under the assumption of uniform priors, write down the formula for the MAP decision rule and determine geometrically the decision regions.

1.5 (Q-function). Show that for $x \geq 0$, $Q(x) \leq e^{-\frac{x^2}{2}}$. Can you show that for $x \geq 0$, $Q(x) \leq \frac{1}{\sqrt{2\pi x^2}} e^{-\frac{x^2}{2}}$?

1.6. Let $Z(t)$ be a real-valued Gaussian process with double-sided power spectral density equal to $\frac{N_0}{2}$. Let $\psi_1(t)$ and $\psi_2(t)$ be two orthonormal functions and for $k = 0, 1$ define the random variables $Z_k = \int_{-\infty}^{\infty} Z(t)\psi_k(t)dt$. What is the distribution of (Z_1, Z_2)?

1.7. In this exercise we continue our review of what happens when stationary stochastic processes are *filtered*. Let $X(t)$ and $U(t)$ denote two stochastic processes and let $Y(t)$ and $V(t)$ be the result of passing $X(t)$ respectively $U(t)$ through linear time invariant filters with impulse response $h(t)$ and $g(t)$, respectively. For any pair (X,U) of stochastic processes define the cross-correlation as

$$\mathcal{R}_{XU}(t_1, t_2) = \mathbb{E}[X(t_1)U^*(t_2)],$$

We say that the pair (X,U) is jointly wide sense stationary if each of them is wide sense stationary and if $\mathcal{R}_{XU}(t_1, t_2)$ is a function of the time difference only. In this case we define a cross-power spectrum as the Fourier transform of the cross-correlation function.

Show that if (X,U) are jointly wide sense stationary then so are (Y,V) and that

$$\mathcal{S}_{YV}(f) = \mathcal{S}_{XV}(f)H(f)G^*(f).$$

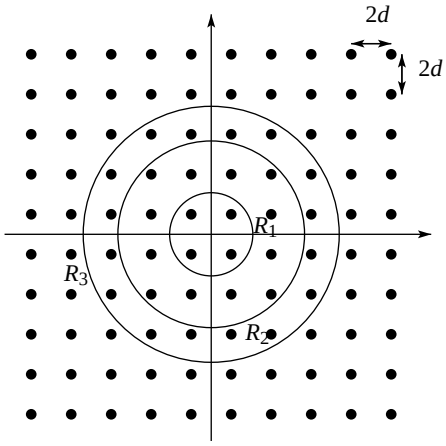
1.8. Show that the cross-correlation function $\mathcal{R}_{XU}(\tau)$ has symmetry

$$\mathcal{R}_{XU}(\tau) = \mathcal{R}^*_{UX}(-\tau)$$

1.9. Let $Z(t)$ be a circularly symmetric stationary zero-mean Gaussian random procress and define $Y(t) = e^{2\pi f_0 t} Z(t)$. Show that $Y(t)$ is also circularly symmetric and stationary.

1.10. In this problem we will be concerned with the design of signal points and in particular the tradeoff between error probability, average energy per signal point and number of signal points.

Consider a grid of points in two dimensions as shown below (only a finite portion is shown).



The points have the form $(d+2dj, d+2dk)$, $j,k \in \mathbb{Z}$, i.e., their vertical and horizontal spacing is $2d$ and they are offset from the origin by d in both the vertical and horizontal direction. Let $\mathcal{S}_{R_1}$ denote the 4 points which are closest to the origin, i.e., the set of points which lie within the disc of radius R_1. In the same manner let $\mathcal{S}_{R_2}$ ($\mathcal{S}_{R_3}$) denote the set of 16 (32) points which are closest to the origin, i.e., the set of points within distance R_2 (R_3) from the origin.

(a) For $\mathcal{S}_{R_1}$ and $\mathcal{S}_{R_2}$ determine the *average energy* of the constellation as a function of d, assuming that all points are equally likely.

Consider now the general problem. Let $\mathcal{S}_R$ denote the set of points which lie within radius R of the origin. We would like to investigate how the number of signal points and the average energy scale with R. To determine these parameters exactly is a difficult problem but we can get a good estimate of these quantities by making the following *continuous approximation*. We will assume that the number of grid points which have modulus (norm) between r and $r+\Delta r$ is equal to $\frac{\pi}{2d^2} r\Delta r$. [We get this approximation by assuming that each grid point "uses" an associated area of $(2d)^2$ (one square in the figure) and by recalling that the area of an annular region of radius r and "width" Δr is equal to $2\pi r\Delta r$.]

(b) Using this approximation, show that the number of grid points within radius R is equal to $\frac{\pi R^2}{4d^2}$ and that the *average* energy of all grid points within radius R is equal to $R^2/2$, assuming that all points are equally likely.

(c) Compare these results with your results in (a) assuming that $R_1 = 2d$ and $R_2 = \sqrt{20}d$.

(d) According to this approximation, by how much do we have to scale R in order to double the number of points, i.e., in order to transmit one extra bit?

Assume now that $\mathcal{S}_R$ are the signal points for a Gaussian hypothesis testing problem where σ^2 is the variance of the noise per dimension. Observe that, under the assumption that $\frac{d}{\sigma}$ is large, an *approximate* expression for the error probability P is given by $P \sim 4Q(\frac{d}{\sigma})$ which is independent of the chosen radius R.

(e) According to this approximation. By how much do we have to scale the energy in order to transmit one extra bit at roughly the same probability of error.

1.11. [Partial Fraction Expansion] Prove the following assertion. The partial fraction expansion is particularly simply when the rational function has only simple poles. Assume we have given

$$H(z) := \frac{F(z)}{G(z)} = \frac{F(z)}{\prod_k (z - z_k)},$$

where the degree of $F(z)$ is less than the degree of $G(z)$ and where are poles z_k are distinct. The claim is that in this case the partial fraction expansion of $H(z)$ is given by

$$H(z) = \sum_k \frac{F(z_k)}{G'(z_k)} \frac{1}{z - z_k} = \sum_k \frac{F(z_k)}{G'(z_k)} \frac{1}{z} \frac{1}{1 - \frac{z_k}{z}}$$

Further, we claim that the *causal* time sequence which possesses this z-transform is

$$h_{n+1} = \sum_{k \geq 0} \frac{F(z_k)}{G'(z_k)} z_k^n$$

What is the the corresponding anticausal time serie? What changes if the degree of $F(z)$ is larger or equal to the degree of $G(z)$?

1.12. In this example we will investigate some basic properties of formal power sums in some more detail. Consider the set of all formal power sums with coefficients in the set F. In class we looked at the case where $F = \{0, 1\}$, the field of two elements. We have seen that we can endow the set of such power sums with some algebraic structure. In particular, we can add, subtract, multiply and for some elements we can even define a division. The set of all formal power sums over F is usually denoted by $F[[D]]$ and is called the *ring* of formal power sums. As a general rule, in $F[[D]]$ all those operations are meaningful which require for the determination of each coefficient of the output only a finite number of (algebraic) operations. We saw how addition, multiplication, and the determination of the multiplicative inverse (if it exists) of a given element all fall in this category.

We will now discuss some more operations which we can perform on elements of $F[[D]]$, we will see how we can use formal powers sums as generating functions to solve problems in the area of enumerative combinatorics and we will investigate the relationship between formal power sums and Taylor series.

1. Assume we are given a sequence $\{x_i\}_{i\geq 0}$. We then say that $x(D) := \sum_{i=0}^{\infty} x_i D^i$ is the generating function of $\{x_i\}_{i\geq 0}$ and we will write $x(D) \leftrightarrow \{x_i\}_{i\geq 0}$.

   (a) If $x(D) \leftrightarrow \{x_i\}_{i\geq 0}$ then what are the generating function of $\{x_{i+1}\}_{i\geq 0}$ and of $\{x_{i+2}\}_{i\geq 0}$?

   (b) Define the *derivative* of a formal power sum $x(D) := \sum_{i=0}^{\infty} x_i D^i$ to be $x'(D) := \sum_{i=0}^{\infty} i x_i D^{i-1}$. If $x(D) \leftrightarrow \{x_i\}_{i\geq 0}$ then what is the formal power sum corresponding to $\{ix_i\}_{i\geq 0}$?

2. Let $f(z)$ be a function such that for some region of convergence $f(z)$ has the Taylor series expansion $f(z) = \sum_{i=0}^{\infty} f_i z^i$. Given a formal power series $x(D) := \sum_{i=0}^{\infty} f_i D^i$ we will then also write $x(D) = f(D)$. Using this notation what would you write for $x_1(D) := \sum_{i=0}^{\infty} \frac{D^i}{i!}$ and for $x_2(D) := \sum_{i=0}^{\infty} (-1)^i \frac{D^{2i}}{(2i)!}$? How about $x_1'(D)$ and $x_2'(D)$? Any comments?

3. Let $\mathbb{F} = \mathbb{R}$ and consider the recurrence $a_{i+2} = a_{i+1} + a_i$, $(i \geq 0; a_0 = a_1 = 1)$. Define the formal power sum $a(D) := \sum_{i=0}^{\infty} a_i D^i$ and use it to solve this recursion. You hopefully got an answer of the form $a(D) = \frac{p(D)}{q(D)}$, where $p(D), q(D) \in F[D]$. Find now a_0, a_1, a_2 and a_3 by *formally* finding the first four coefficients of the resulting power sum. Note: Do not use the recursion for that, start with $a(D)$ and use only algebraic operations. Now use a *partial fraction expansion* to write $a(D)$ as a sum of rational terms each of which has only one pole. Can this procedure be again defined in a purely formal way, i.e., only using algebraic operations but making no use of any analytic properties? Finally, use this partial fraction expansion to give an expression of the coefficients a_i in a somewhat more explicit form.

4. Let $\mathbb{F} = \mathbb{R}$ and consider the recurrence $(n+1)a_{n+1} = 3a_n + 1$, $(n \geq 0; a_0 = 1)$. Define the formal power sum $a(D) := \sum_{i=0}^{\infty} a_i D^i$ and use it to solve this recursion.

5. Let $x(D), y(D) \in F[[D]]$. We are interested in *compositions* of formal power sums. Can you find a meaningful definition for the expression $y(x(D))$? Does such an expression always make sense. Using your findings: Does $e^{e^D - 1}$ have a well defined formal power series? How about e^{e^D}?

1.13. Define the two formal power sums $x(D) := \sum_{i=0}^{\infty} \frac{1}{i!} D^i$ and $y(D) := -\sum_{i=1}^{\infty} \frac{(-1)^i}{i} D^i$, where all coefficients are over $\mathbb{R}$.

1. Do $\frac{1}{x(D)}$ and $x(y(D))$ exist? If so, determine their first three coefficients.

2. Show that $x'(D) = x(D)$ and $y'(D) = \frac{1}{1+D}$, where all operations are interpreted formally. [Recall that the formal derivative of a formal power sum $z(D) := \sum_{i=0}^{\infty} z_i D^i$ is equal to $\sum_{i=0}^{\infty} i z_i D^{i-1} = \sum_{i=0}^{\infty} (i+1) z_{i+1} D^i$.]

3. Find functions $f(D)$ and $g(D)$ such that $x(D)$ and $y(D)$ are their respective Taylor series around zero. [HINT: You might recognize the functions $f(D)$ and $g(D)$ from their respective Taylor series $\sum_{i=0}^{\infty} \frac{1}{i!} D^i$ and $-\sum_{i=1}^{\infty} \frac{(-1)^i}{i} D^i$ directly. If not, observe from above that $f(D)$ and $g(D)$ fulfil the equations $f'(D) = f(D)$ and $g'(D) = \frac{1}{1+D}$.]

4. Use the above functions to write down $\frac{1}{x(D)}$ and $x(y(D))$ explicitly as formal power sums.

1.14. In this exercise we will review the Euclidean algorithm, an efficient algorithm to calculate the *greatest common divisor*. Recall that $\gcd(a,b)$, where a and b are integers is the largest integer c such that c divides a and c divides b. We have the following elementary properties of the gcd.

Fact 1. [Basic properties of the gcd]

(i) $\gcd(a,b) = \gcd(b,a)$,

(ii) $gcd(a,0) = a$.

(iii) $gcd(a,b) = \gcd(a,b+ca)$.

This gives us a way to calculate the gcd by choosing c so that $\gcd(a,b+ca)$ is "simpler" than $\gcd(a,b)$. It is common to choose c in such a way that $b+ca$ is as small as possible. In particular assume that $|b| > |a|$ and let c be an integer such that $b = ca + r$ with the *remainder* r such that $0 \leq r < a$. Then $\gcd(a,b) = \gcd(a,b-ca) = \gcd(a,r)$. Now repeat this procedure until one of the arguments is zero. The remaining non-zero argument is then the sought after greatest common divisor. Use this algorithm to calculate the greatest common divisor of 1573 and 308. Can you extend this algorithm to find integers a and b such that $\gcd(1573,308) = a1573 + b308$?

Now note that the same algorithm works also for polynomials. Look at the set of polynomials over some field F (think of F as the reals or complex numbers or the binary field). For polynomials, the *degree* plays the role of the absolute value of integers. Polynomials of degree zero are called the *scalars*. A polynomial is called *monic* if its leading coefficient is one. As in the case of integers we can divide, i.e., given two polynomials $a(D)$ and $b(D)$ with $\deg a \geq \deg(b)$ we can find unique polynomials $c(D)$ and $r(D)$ such that $a(x) = b(D)c(D) + r(D)$ with $\deg(D) < \deg(b)$. A polynomial which can not be written as the product of two other polynomials (of degree at least one) is called *irreducible*. An irreducible polynomial is the equivalent to a *prime* in the realm of integers. Be aware that irreducibility depends on the field F over which we regard the polynomial. E.g.,

$1+x^2$ is irreducible over the reals but does factor into two linear terms over the complex numbers. Then, as for integers, we have a *unique factorization* theorem for polynomial, i.e., a given polynomial can be factored in a unique way into irreducible monic factors and a scalar. The greatest common division of two polynomials $a(D)$ and $b(D)$ is defined to be that unique monic polynomial $c(D)$ which divides both polynomials and has the largest degree of all such polynomials. The Euclidean algorithm can be extended in a straightforward way to the setting of polynomials to determine the gcd in an efficient way.

Use the Euclidean algorithm to calculate the greatest common divisor of the following two pairs of polynomials over the reals: (x^4-x^3+x-1, x^2-x+1) and (x^4-x^2+x-1, x^3-x^2+1).

1.15. [Wilf's Snake Oil Method] The following exercise deals with a simple method that can often help you do find explicit expressions for seamingly complex sums. Consider the sum

$$f(n) = \sum_{k\geq 0} \binom{k}{n-k}$$

We want to find an "explicit" formula for $f(n)$. The trick is to consider the generating function $F(x) := \sum_n f(n)x^n$ instead, i.e., rather than asking for the solution for a particular n we would like to solve the problem for all n simultaneously! This seems to make the problem if anything harder not easier. But now we have a double sum and as we will see, by exchanging the order of summation, we can actually solve the problem.

$$\begin{aligned}
F(x) &= \sum_{n\geq 0} f(n)x^n \\
&= \sum_{n\geq 0} \left(\sum_{k\geq 0} \binom{k}{n-k} \right) x^n \\
&= \sum_{k\geq 0} \left(\sum_{n} \binom{k}{n-k} x^n \right) \\
&= \sum_{k\geq 0} \left(x^k \sum_{n} \binom{k}{n-k} x^{n-k} \right) \\
&= \sum_{k\geq 0} \left(x^k \sum_{m} \binom{k}{m} x^m \right) \\
&= \sum_{k\geq 0} x^k (1+x)^k \\
&= \frac{1}{1-x-x^2} \\
&= \frac{1}{x_+ - x_-} \left(\frac{1}{1-x_+ x} - \frac{1}{1-x_- x} \right) \\
&= \sum_{n\geq 0} \frac{1}{\sqrt{5}} \left(x_+^n - x_-^n \right) x^n,
\end{aligned}$$

where we have used the binomial identity and a standard partial fraction expansion and where we defined $x_\pm = (1 \pm \sqrt{5})/2$. We therefore have our sought after answer

$$\sum_{k\geq 0} \binom{k}{n-k} = \frac{1}{\sqrt{5}} \left(x_+^n - x_-^n\right).$$

Apply the same trick to the scary looking summation

$$f(n,m) := \sum_{k\geq 0} \binom{n+k}{m+2k} \binom{2k}{k} \frac{(-1)^k}{k+1}, \;\; m,n \geq 0,$$

Hints: Consider m fixed and sum again over n! You might find it helpful to know that

$$\sum_{n\geq 0} \binom{n}{k} x^n = \frac{x^k}{(1-x)^{k+1}}$$

and that

$$\sum_{n\geq 0} \frac{1}{n+1} \binom{2n}{n} x^n = \frac{1}{2x}(1 - \sqrt{1-4x}).$$

2

TRANSMISSION OVER LINEAR TIME-INVARIANT CHANNELS

So far we know how to transmit information over an *ideal* passband/baseband channel with AWGN. In this chapter we will consider a more general model. We will consider *pulse amplitude modulated* signals transmitted over a linear time-invariant channel with AWGN.

1. MAXIMUM LIKELIHOOD SEQUENCE ESTIMATOR: VITERBI ALGORITHM

Let

$$x_E(t) = \sum_{n=0}^{N-1} x_n \psi(t - nT), \tag{2.1}$$

be the pulse amplitude modulated baseband signal, where x_n takes elements from some finite (complex-valued) set $\mathcal{X}$. Let $h_E(t)$ denote the baseband equivalent channel impulse response. Then the received signal is given by (see (1.13))

$$y_E(t) = x_E(t) * \frac{1}{\sqrt{2}} h_E(t) + Z(t) = \sum_{n=0}^{N-1} x_n g_E(t - nT) + Z(t), \tag{2.2}$$

where $g_E(t) := \psi(t) * \frac{1}{\sqrt{2}} h_E(t)$, and where $Z(t)$ is a complex circularly-symmetric Gaussian process with power spectral density equal to N_0. Note that this is again a pulse amplitude modulated signal but that the pulse shape now incorporates also the effect of the channel. Since the channel is usually not under the control of the system designer, the pulse $g_E(t)$ will, in general, not satisfy the Nyquist criterion. We will therefore have *intersymbol-interference* (ISI).

Although we assume that the channel is not known for the *design phase* of the system, we will assume that the *receiver knows the channel (perfectly)*. We will give a justification of this assumption at some later point where we will show how to perform the necessary *channel estimation task*.

We see from (2.1) that all possible transmitted points lie in some finite dimensional subspace. We can find a basis for this subspace by applying a Gram-Schmidt procedure to the set $\{g_E(t),\ldots,g_E(t-(N-1)T)\}$. Under the assumption of uniform priors we now know that the optimal receiver can first project the received signal $y_E(t)$ onto this subspace, call the result $\tilde{y}_E(t)$, and then find that point $\tilde{x}_E(t) := \sum_{n=1}^{N-1} \tilde{x}_n g_E(t-nT)$ out of all $|\mathcal{X}|^N$ such points which is *closest* to $\tilde{y}_E(t)$, see (1.1). For any two (complex-valued) functions $a(t)$ and $b(t)$ define $\langle a(t),b(t)\rangle := \int_{-\infty}^{\infty} a(t)b^*(t)dt$. Since

$$\begin{aligned}\|\tilde{x}_E(t)-\tilde{y}_E(t)\|^2 &= \|\tilde{x}_E(t)\|^2+\|\tilde{y}_E(t)\|^2-\langle\tilde{x}_E(t),\tilde{y}_E(t)\rangle-\langle\tilde{y}_E(t),\tilde{x}_E(t)\rangle\\ &= \|\tilde{x}_E(t)\|^2+\|\tilde{y}_E(t)\|^2-2\,\mathrm{Re}\langle\tilde{y}_E(t),\tilde{x}_E(t)\rangle,\end{aligned}$$

and since $\|\tilde{y}_E(t)\|^2$ is common to all such terms, we see that rather than minimizing the Euclidean distance we can maximize the expression

$$2\,\mathrm{Re}\langle\tilde{y}_E(t),\tilde{x}_E(t)\rangle-\|\tilde{x}_E(t)\|^2,$$

see (1.2). But $\langle\tilde{y}_E(t),\tilde{x}_E(t)\rangle=\langle y_E(t),\tilde{x}_E(t)\rangle$, so that we can maximize the quantity

$$2\,\mathrm{Re}\langle y_E(t),\tilde{x}_E(t)\rangle-\|\tilde{x}_E(t)\|^2$$

instead. (Why did we bother to introduce the projection $\tilde{y}_E(t)$?) Recall that we have in total $|\mathcal{X}|^N$ signals. Therefore at first it appears that we need to perform $|\mathcal{X}|^N$ inner products. This would of course we prohibitively complex. As we will see now, we can do much better.

Explicitly, we have

$$\begin{aligned}&2\,\mathrm{Re}\langle y_E(t),\tilde{x}_E(t)\rangle-\|\tilde{x}_E(t)\|^2\\ =\;& 2\,\mathrm{Re}\left\{\int y_E(t)\sum_{n=0}^{N-1}\tilde{x}_n^* g_E^*(t-nT)dt\right\}-\int\sum_{n=0}^{N-1}\sum_{m=0}^{N-1}\tilde{x}_n^*\tilde{x}_m g_E^*(t-nT)g_E(t-mT)dt\\ =\;& 2\,\mathrm{Re}\left\{\sum_{n=0}^{N-1}\tilde{x}_n^*\int y_E(t)g_E^*(t-nT)dt\right\}-\sum_{n=0}^{N-1}\sum_{m=0}^{N-1}\tilde{x}_n^*\tilde{x}_m\int g_E^*(t-nT)g_E(t-mT)dt\\ =\;& 2\,\mathrm{Re}\left\{\sum_{n=0}^{N-1}\tilde{x}_n^* y_n\right\}-\sum_{n=0}^{N-1}\sum_{m=0}^{N-1}\tilde{x}_n^*\tilde{x}_m\mathcal{R}_g(n-m),\end{aligned}$$

where

$$\mathcal{R}_g(k) := \int g_E(t)g_E^*(t-kT)dt,$$

and where we defined

$$y_n := \int y_E(t)g_E^*(t-nT)dt.$$

Note that

$$\mathcal{R}_g(-k) = \mathcal{R}_g^*(k).$$

Assume now that $g_E(t)$ has *finite support*, i.e., $g_E(t) = 0$ for $t \geq LT$ so that $\mathcal{R}_g(k) = 0$ for $|k| \geq L$. In this case we have

$$\begin{aligned}
&= 2\,\mathrm{Re}\left\{\sum_{n=0}^{N-1} \tilde{x}_n^* y_n\right\} - \underbrace{\sum_{n=0}^{N-1}\sum_{m=0}^{N-1} \tilde{x}_n^* \tilde{x}_m \mathcal{R}_g(n-m)}_{\text{real valued: conjugates } m \leftrightarrow n} \\
&= 2\,\mathrm{Re}\left\{\sum_{n=0}^{N-1} \tilde{x}_n^* y_n\right\} - \mathcal{R}_g(0)\sum_{n=0}^{N-1} x_n^* x_n - 2\,\mathrm{Re}\left\{\sum_{n=0}^{N-1} \tilde{x}_n^* \sum_{m<n} \tilde{x}_m \mathcal{R}_g(n-m)\right\} \\
&= 2\,\mathrm{Re}\left\{\sum_{n=0}^{N-1} \tilde{x}_n^* y_n\right\} - \mathcal{R}_g(0)\sum_{n=0}^{N-1} x_n^* x_n - 2\,\mathrm{Re}\left\{\sum_{n=0}^{N-1} \tilde{x}_n^* \sum_{m=1}^{L-1} \mathcal{R}_g(m) \tilde{x}_{n-m}\right\} \\
&= \sum_{n=0}^{N-1} 2\,\mathrm{Re}\left\{\tilde{x}_n^*\left(y_n - \frac{1}{2}\mathcal{R}_g(0)\tilde{x}_n - \sum_{m=1}^{L-1} \mathcal{R}_g(m)\tilde{x}_{n-m}\right)\right\},
\end{aligned}$$

where we assumed that we defined $\tilde{x}_n := 0$ for $n < 0$ and $n \geq N$. So we see that the set of inner products $\{y_0, \ldots, y_{N-1}\}$ constitutes a sufficient statistic. We will now see that we can employ the *Viterbi algorithm* which you encountered already in the decoding of *convolutional codes* to perform this maximization in an efficient manner.

We need to find

$$\begin{aligned}
&\mathrm{argmax}_{\tilde{x}_0,\ldots,\tilde{x}_{N-1}} \sum_{n=0}^{N-1} 2\,\mathrm{Re}\left\{\tilde{x}_n^*\left(y_n - \frac{1}{2}\mathcal{R}_g(0)\tilde{x}_n - \sum_{m=1}^{L-1} \mathcal{R}(m)\tilde{x}_{n-m}\right)\right\} \\
= \;&\mathrm{argmax}_{\tilde{x}_0,\ldots,\tilde{x}_{N-1}} \sum_{n=0}^{N-1} m(y_n; \tilde{x}_n; \tilde{x}_{n-L+1}, \ldots, \tilde{x}_{n-1}) \\
= \;&\mathrm{argmax}_{\tilde{x}_0,\ldots,\tilde{x}_{N-1}} \sum_{n=0}^{N-1} m(y_n; \tilde{x}_n; \sigma_n),
\end{aligned}$$

where we defined $\sigma_n := (\tilde{x}_{n-L+1}, \ldots, \tilde{x}_{n-1})$. We call σ_n the *state* at time n. Note that the metric is the sum of N parts, where the n-th part depends on the n-th received value y_n, the n-th conjectured transmitted value $\tilde{x}_n$, as well as the state at time n.

Consider first the case $L = 1$. In this case the state is empty and the individual terms have the form $m(y_n; \tilde{x}_n)$. Therefore

$$\max_{\tilde{x}_0,\ldots,\tilde{x}_{N-1}} \sum_{n=0}^{N-1} m(y_n; \tilde{x}_n) = \sum_{n=0}^{N-1} \max_{\tilde{x}_n} m(y_n; \tilde{x}_n).$$

In words, the optimal detector can proceed symbol-by-symbol wise. This is of course expected, since in this case we do not have inter-symbol-interference.

Assume now that $L > 1$, so that the state is now non-trivial. In this case a symbol-by-symbol optimization is not optimal. But for any $0 \leq l \leq N-1$ we have:

$$\begin{aligned} & \max_{\tilde{x}_0,\ldots,\tilde{x}_{N-1}} \sum_{n=0}^{N-1} m(y_n;\tilde{x}_n;\sigma_n) \\ = & \max_{\sigma_l=\alpha} \left\{ \max_{\tilde{x}_0,\ldots,\tilde{x}_{l-1}:\sigma_l=\alpha} \sum_{n=0}^{l-1} m(y_n;\tilde{x}_n;\sigma_n) + \max_{\tilde{x}_l,\ldots,\tilde{x}_{N-1}:\sigma_l=\alpha} \sum_{n=l}^{N-1} m(y_n;\tilde{x}_n;\sigma_n) \right\} \end{aligned}$$

In words, conditioned that we pass through a certain state at time l the "future" and "past" are independent. Even a single application of this fact results in a large savings. Assume we pick $l = N/2$. In this case for each possible state the evaluation of the above maximization has complexity $2|\mathcal{X}|^{N/2}$ so that the total complexity is roughly equal to $2|\mathcal{X}|^{N/2+L-1}$. Applying this trick repeatedly results in the Viterbi algorithm. The Viterbi algorithm has a nice graphical representation in terms of the so-called *trellis diagram*. All these concepts are probably most easily explained in terms of an example.

Example 6. Consider antipodal transmission and assume that

$$\mathcal{R}_g(0) = 1, \ \mathcal{R}_g(1) = \frac{1}{2}, \ \mathcal{R}_g(k) = 0, k \geq 2.$$

Let $N = 5$ and assume that the received vector is equal to

$$(y_0, y_1, y_2, y_3, y_4) = (0.7, -2.3, -0.5, 0.4, 2.5).$$

Recall that in general the state at time n was defined as $\sigma_n := (\tilde{x}_{n-L-1}, \ldots, \tilde{x}_{n-1})$ so that for our case we have $\sigma_n = (\tilde{x}_{n-1})$. Note that for each possible sequence $\tilde{x}_0, \ldots, \tilde{x}_{N-1}$ there is a unique sequence $\sigma_1, \ldots, \sigma_N = (\tilde{x}_0), \ldots, (\tilde{x}_{N-1})$ and vice versa. Consider the following graph, usually referred to as the *trellis-diagram*. Nodes correspond to states and edges correspond to transmitted and received values. The leftmost node corresponds to the *initial state* which for our case is equal to zero since we assumed that $x_n = 0$ for $n < 0$. The two nodes at time one correspond to the two possible states after the first bit has been transmitted, namely the states $+1$ and -1. The two edges connecting the initial state to these two states at

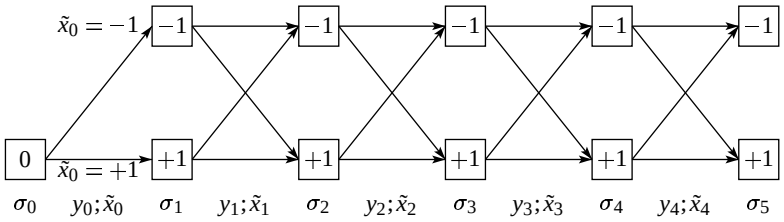


Figure 2.1: The trellis diagram for antipodal transmission, $L = 2$ and $N = 5$.

time one correspond to the two possible hypothesis, namely that $\tilde{x}_0 = \pm 1$. Note that each possible state sequence (and hence each possible sequence of transmitted bits) corresponds in a unique way to a single path through this trellis.

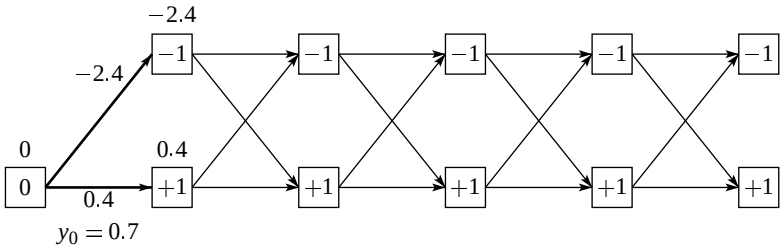
We want to perform the maximization

$$\max_{\tilde{x}_0,\ldots,\tilde{x}_{N-1}} \sum_{n=0}^{N-1} m(y_n;\tilde{x}_n;\sigma_n).$$

Look at the first term, i.e., $m(y_0;\tilde{x}_0;\sigma_0)$. Since $\sigma_0 = (0)$ we have

$$m(y_0;\tilde{x}_0;\sigma_0) = \begin{cases} 2\cdot 1\cdot\{0.7-\frac{1}{2}\cdot 1-\frac{1}{2}\cdot 0\} = 0.4, & \tilde{x}_0 = +1, \\ 2\cdot(-1)\cdot\{0.7-\frac{1}{2}\cdot(-1)-\frac{1}{2}\cdot 0\} = -2.4, & \tilde{x}_0 = -1. \end{cases}$$

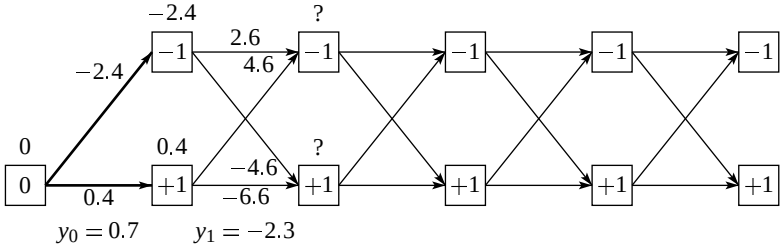
Assume now we label the edges emanating from the initial states with $m(y_0;\tilde{x}_0;\sigma_0)$. Assume further, that we label states with the *accumulated* metrics, where the accumulation is over the labels of the edges along the (shortest) path from the initial state. More precisely, we label the single node corresponding to $\sigma_0 = (0)$ with zero, we label the node corresponding to $\sigma_1 = +1$ with $m(y_0;\tilde{x}_0 = +1;\sigma_0) = 0.4$ and we label the node corresponding to $\sigma_1 = -1$ with $m(y_0;\tilde{x}_0 = -1;\sigma_0) = -2.4$.



As a next step determine $m(y_1;\tilde{x}_1;\sigma_1)$. We get

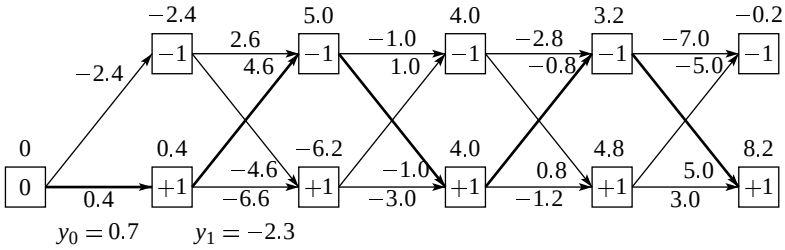
$$m(y_1;\tilde{x}_1;\sigma_1) = \begin{cases} 2\cdot 1\cdot\{-2.3-\frac{1}{2}\cdot 1-\frac{1}{2}\cdot 1\} = -6.6, & \tilde{x}_1 = +1;\sigma_1 = (+1) \\ 2\cdot(-1)\cdot\{-2.3-\frac{1}{2}\cdot(-1)-\frac{1}{2}\cdot 1\} = 4.6, & \tilde{x}_1 = -1;\sigma_1 = (+1), \\ 2\cdot 1\cdot\{-2.3-\frac{1}{2}\cdot 1-\frac{1}{2}\cdot(-1)\} = -4.6, & \tilde{x}_1 = +1;\sigma_1 = (-1) \\ 2\cdot(-1)\cdot\{-2.3-\frac{1}{2}\cdot(-1)-\frac{1}{2}\cdot(-1)\} = 2.6, & \tilde{x}_1 = -1;\sigma_1 = (-1). \end{cases}$$

Adding these edge labels to our trellis we arrive at the following picture.



We now get to the crucial point of the Viterbi algorithm. Note that to each of the (two) states at time two there are *two* distinct paths. E.g., to the state $\sigma_2 = +1$

we arrive from the initial state by either the hypothesis $\tilde{x}_0 = +1, \tilde{x}_1 = +1$ or by the hypothesis $\tilde{x}_0 = -1, \tilde{x}_1 = +1$. The first path, call it $\mathtt{p}_1$, has an accumulated metric of $m(y_0; \tilde{x}_0 = +1; \sigma_0 = (0)) + m(y_1; \tilde{x}_1 = +1; \sigma_1 = (+1)) = -6.2$ the second path, call it $\mathtt{p}_2$, has an accumulated metric of $m(y_0; \tilde{x}_0 = -1; \sigma_0 = (0)) + m(y_1; \tilde{x}_1 = +1; \sigma_1 = (-1)) = -7.0$. We can now argue as follows: any complete path $\mathtt{p}$ through the trellis which passes through state $\sigma_2 = (+1)$ must have an initial portion equal either to $\mathtt{p}_1$ or $\mathtt{p}_2$. The accumulated metric of this path is composed of the metric of the initial segment plus the metric of the remaining segment. *But this additional term does not depend on the initial segment given that we pass through the said state!* We conclude that the optimal path, if it passes through state $\sigma_2 = (+1)$, must have initial segment $\mathtt{p}_1$. Therefore we can label $\sigma_2 = (+1)$ with -6.2. Note that this label is equal to the label of $\sigma_1 = (+1)$ plus the edge which connects $\sigma_1 = (+1)$ to $\sigma_2 = (+1)$. Continuing this procedure we arrive at the following picture.



By *tracing back* the *surviving* path we can read off the sequence $\tilde{x}_0, \ldots, \tilde{x}_{N-1}$ which is closest to the received point. We get in our case the estimated sequence $+1-1+1-1+1$.

2. THE EQUIVALENT DISCRETE TIME CHANNEL

We have seen from the previous example that the samples $y_0, \ldots, y_{N-1}$, where $y_n = \int y(t) g_E^*(t - nT)dt$, constitute a sufficient statistic, i.e., that we can make an optimal decision based solely on these quantities. Therefore, rather than looking at the continuous time channel

$$y_E(t) = \sum_{n=0}^{N-1} x_n g_E(t - nT) + Z(t),$$

we can focus on the *equivalent* discrete time channel

$$y_n = \sum_k \mathcal{R}_g(k) x_{n-k} + z_n, \tag{2.3}$$

where $z_n := \int Z(t) g_E^*(t-nT)dt$. Note, however that the noise is *correlated*. We have

$$\begin{aligned}
\mathcal{R}_z(k) &= \mathbb{E}[z_n z_{n-k}^*] \\
&= \mathbb{E}\left[\left(\int Z(t) g_E^*(t-nT)dt\right)\left(\int Z^*(\tau) g_E(\tau-(n-k)T)d\tau\right)\right] \\
&= \int\int g_E^*(t-nT) g_E(\tau-(n-k)T)\mathbb{E}[Z(t)Z^*(\tau)]dtd\tau \\
&= \int\int g_E^*(t-nT) g_E(\tau-(n-k)T)N_0\delta(t-\tau)dtd\tau \\
&= N_0\int g_E^*(t-nT) g_E(t-(n-k)T)dt \\
&= N_0\int g_E(t) g_E^*(t-kT)dt \\
&= N_0\mathcal{R}_g(k).
\end{aligned}$$

2.1 THE WHITENING FILTER

It is much more convenient to deal with white noise. We will therefore now see how we can *filter* (2.3) such that the resulting signal has a noise component which is white.

Let's first recall what happens if we send a discrete time WSS process z_n through a filter with impulse response h_n. Define

$$\mathcal{R}_z(k) := \mathbb{E}[z_n z_{n-k}^*],$$

and let w_n be the output of the filter, i.e.,

$$w_n := \sum_m h_m z_{n-m}.$$

Then we have

$$\begin{aligned}
\mathcal{R}_w(k) &:= \mathbb{E}[w_n w_{n-k}^*] \\
&= \mathbb{E}\left[\left(\sum_m h_m z_{n-m}\right)\left(\sum_l h_l^* z_{n-k-l}^*\right)\right] \\
&= \sum_m\sum_l h_m h_l^*\mathbb{E}[z_{n-m}z_{n-k-l}^*] \\
&= \sum_m\sum_l h_m h_l^*\mathcal{R}_z(k-m+l) \\
&= \sum_m\sum_l h_m h_{-l}^*\mathcal{R}_z(k-m-l) \\
&= \sum_m h_m\sum_l \left(h_{-l}^*\mathcal{R}_z((k-m)-l)\right) \\
&= \left(h_n * h_{-n}^* * \mathcal{R}_z(n)\right)(k). \qquad (2.4)
\end{aligned}$$

As in the case of continuous processes, we can define the *spectrum* of a discrete time WSS process as the (z or Fourier) transform of its autocorrelation function. From Appendix 5.1 we get

$$\mathcal{S}(z) := \sum_k \mathcal{R}(k) z^{-k},\ \mathcal{S}(e^{2\pi jf}) := \sum_k \mathcal{R}(k) z^{-k}|_{z=e^{2\pi jf}},\ \mathcal{R}(k) = \int_0^1 \mathcal{S}(e^{2\pi jf}) e^{2\pi jfk} df.$$

Recall that the convolution of two time domain signals corresponds to the multiplication of their respective transforms (z-Transform or DTFT). Therefore we get from (2.4) the relationships

$$\mathcal{S}_w(z) = H(z)H^*(1/z^*)\mathcal{S}_z(z), \quad \mathcal{S}_w(e^{2\pi jf}) = H(e^{2\pi jf})H^*(e^{2\pi jf})\mathcal{S}_z(e^{2\pi jf}). \quad (2.5)$$

Assume now that $\mathcal{R}_z(k) = 0$ for $k \geq L$ and that $\mathcal{R}_z(L-1) \neq 0$. Consider the corresponding spectrum $\mathcal{S}_z(z)$,[1]

$$\mathcal{S}_z(z) = \sum_{n=-(L-1)}^{L-1} \mathcal{R}_z(k) z^{-n}.$$

Note that $p_+(z) := \mathcal{S}_z(z) z^{(L-1)}$ is a polynomial in z of degree exactly $2(L-1)$. Further, since $\mathcal{R}_z(k) = \mathcal{R}_z^*(-k)$ it follows from the conjugacy rule of the z-transform, see (1.3), that $\mathcal{S}_z(z) = \mathcal{S}_z^*(1/z^*)$. We conclude that the $2(L-1)$ roots of $\mathcal{S}_z(z)$ have the symmetry that if ρ is a root then so is $1/\rho^*$.[2] We say that ρ and $1/\rho^*$ are *conjugate-symmetric*. This symmetry relationship is shown in Fig. B.1. We claim

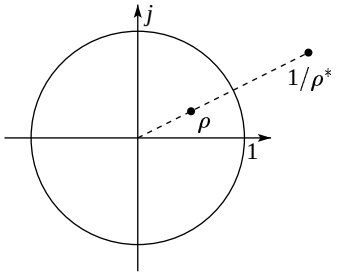


Figure 2.2: A conjugate-symmetric pair ρ and $1/\rho^*$.

that $\mathcal{S}_z(z)$ can be factored as

$$\mathcal{S}_z(z) = F(z)F^*(1/z^*),$$

where $F(z)$ is a polynomial in z^{-1}. To see this claim note that $p_+(z)$ has only non-zero roots and that any non-zero root of $p_+(z)$ is also a root of $\mathcal{S}_z(z)$. Since

[1] This is an unfortunate double use of symbols: z here denotes the random variable as well as the symbol for the z-Transform!

[2] $(L-1)$ out of the $2(L-1)$ poles are at $z = 0$ and the remaining ones are at $z = \infty$.

$p_+(z)$ is a polynomial in z of degree $2(L-1)$ with conjugate roots it follows that it has a factorization of the form

$$p_+(z) = A\prod_{i=1}^{L-1}(1-\frac{1}{\rho_i}z)(z-\frac{1}{\rho_i^*}).$$

Therefore

$$\mathcal{S}_z(z) = \frac{p_+(z)}{z^{-(L-1)}} = A\prod_{i=1}^{L-1}(1-\frac{1}{\rho_i}z)(1-\frac{1}{\rho_i^*}z^{-1}),$$

proving the claim.

Assume now that we pass y_n through a filter with spectrum $\frac{\sqrt{N_0}}{F^*(1/z^*)}$. The result, call it r_n, will have the form

$$r_n := \sum_l f_l x_{n-l} + w_n.$$

Let's first consider the noise sequence w_n. From (2.5) its spectrum is given by

$$\mathcal{S}_w(z) = H(z)H^*(1/z^*)\mathcal{S}_z(z) = \frac{\sqrt{N_0}}{F^*(1/z^*)}\frac{\sqrt{N_0}}{F(z)}F(z)F^*(1/z^*) = N_0,$$

so that we see that the process is now white! Therefore the filter $\frac{\sqrt{N_0}}{F^*(1/z^*)}$ is called a *whitening* filter. Next lets consider the impulse response f_n. It is the result of convolving $\mathcal{R}_g(n)$ with the impulse response corresponding to the filter $\frac{\sqrt{N_0}}{F^*(1/z^*)}$. Since $F(z)F^*(1/z^*) = S_z(z) = N_0 S_g(z)$ it follows that

$$\frac{S_g(z)\sqrt{N_0}}{F^*(1/z^*)} = \frac{1}{\sqrt{N_0}}F(z).$$

Therefore f_n is seen to be the inverse transform of $\frac{1}{\sqrt{N_0}}F(z)$!

We still have a large degree of freedom in choosing $F(z)$. This degree of freedom stems from our choice in assigning the roots of $\mathcal{S}_z(z)$ to either $F(z)$ or $F^*(1/z^*)$. We would like our filter $F(z)$ to be *causal, stable and minimum phase*. Causality and stability require that all poles of $F(z)$ are within the unit circle. This is always true since $F(z)$ is a polynomial in z^{-1} and so all its poles are at $z=0$. In order for $F(z)$ to be minimum phase we require that (besides all poles also) all zeros of $F(z)$ are within the unit circle. This can always be accomplished since as we saw beforehand all zeros of $\mathcal{S}_w(z)$ come in conjugate-symmetric pairs. Note though that with this choice, all the zeros and poles of $F^*(1/z^*)$ are *outside* the unit circle, so that $F^*(1/z^*)$ is stable but *anticausal*. For implementation purposes this is not of big concern since we can always introduce a sufficiently large delay to implement such a filter.

We summarize: By passing y_n through a cleverly chosen filter we can (i) whiten the noise and (ii) make the equivalent impulse response f_n to be causal.

Example 7. Assume we have $\mathcal{R}_g(0) = 1 + |a|^2$, $\mathcal{R}_g(1) = a$, $\mathcal{R}_g(k) = 0$ for $k \geq 2$. Then

$$\mathcal{S}_z(z) = N_0(a^* z + (1 + (|a|^2) + az^{-1}) = N_0(az^{-1} + 1)(a^* z + 1) = N_0(\frac{1}{a^*}z^{-1} + 1)(\frac{1}{a}z + 1)|a|^2.$$

If $|a| < 1$, then we pick $F(z) = \sqrt{N_0}(1 + az^{-1})$, whereas if $|a| > 1$ then we pick $F(z) = \sqrt{N_0}(a^*(1 + \frac{1}{a^*}z^{-1}))$. In the first case we get

$$f_n := \begin{cases} 0, & n < 0, \\ 1, & n = 0, \\ a, & n = 1, \\ 0, & n \geq 2. \end{cases}$$

and the resulting channel is

$$r_n = x_n + ax_{n-1} + w_n.$$

□

Remark: We cheated slightly in our derivation above by invoking arguments in the frequency domain (which assumes that all processes are stationary) while at the same time assuming that the transmitted sequence was of finite length.

2.2 THE VITERBI ALGORITHM FOR THE EQUIVALENT DISCRETE TIME CHANNEL

Assume now that we have the following discrete-time channel model.

$$y_n = \sum_{l=0}^{L-1} h_l x_{n-l} + z_n, \; n = 0, \ldots, N-1,$$

where all quantities are either real or complex valued, x_i takes values in some discrete subset of $\mathbb{R}$ or $\mathbb{C}$, h represents the channel impulse response which is causal and of length L and the z_n are i.i.d. random variables with density $p(z)$, not necessarily Gaussian. We have seen in the previous section *one way* in which such a channel model might arise.

In Exercise 2.3 we discussed how we can apply the Viterbi algorithm to this case to implement the optimal detector in an efficient manner. The key lies in the

fact that

$$\begin{aligned}
& \max_{\tilde{x}_0,\ldots,\tilde{x}_{N-1}} p(y_0,\ldots,y_{N-1}|\tilde{x}_0,\ldots,\tilde{x}_{N-1}) \\
= & \max_{\tilde{x}_0,\ldots,\tilde{x}_{N-1}} p(z_0 = y_0 - \sum_{l=0}^{L-1} h_l \tilde{x}_{0-l},\ldots,z_{N-1} = y_{N-1} - \sum_{l=0}^{L-1} h_l \tilde{x}_{N-1-l}) \\
= & \max_{\tilde{x}_0,\ldots,\tilde{x}_{N-1}} \prod_{n=0}^{N-1} p(z_n = y_n - h_0\tilde{x}_n - \sum_{l=1}^{L-1} h_l \tilde{x}_{n-l}) \\
= & \max_{\tilde{x}_0,\ldots,\tilde{x}_{N-1}} \prod_{n=0}^{N-1} e^{m(y_n;\tilde{x}_n;\sigma_n)}.
\end{aligned}$$

If we apply the logarithm to both sides of the above equation, then we see that we can label the edges of the trellis with the quantities

$$m(y_n;\tilde{x}_n;\sigma_n) = \ln p(z_n = y_n - h_0\tilde{x}_n - \sum_{l=1}^{L-1} h_l \tilde{x}_{n-l}),$$

and then employ the Viterbi algorithm to find the most likely sequence. Alternatively we can label the edges directly with

$$e^{m(y_n;\tilde{x}_n;\sigma_n)} = p(z_n = y_n - h_0\tilde{x}_n - \sum_{l=1}^{L-1} h_l \tilde{x}_{n-l}),$$

and then modify the Viterbi algorithm to *multiply* the partial metrics rather than to *add* them.

It is easy to see that if we have a prior of the form

$$\Pr(\tilde{x}_0,\ldots,\tilde{x}_{N-1}) = \prod_{n=0}^{N-1} \Pr(\tilde{x}_n),$$

then such a prior can be taken into account simply by replacing the partial metrics $p(z_n = y_n - h_0\tilde{x}_n - \sum_{l=1}^{L-1} h_l \tilde{x}_{n-l})$ with $p(z_n = y_n - h_0\tilde{x}_n - \sum_{l=1}^{L-1} h_l \tilde{x}_{n-l})\Pr(\tilde{x}_n)$.

2.3 THE BCJR ALGORITHM FOR THE EQUIVALENT DISCRETE TIME CHANNEL

We have seen in the previous two sections how we can use the Viterbi algorithm to find the most probable *sequence* given the observation. Such a detector minimizes the probability of *sequence error.* In some instances it is more natural to minimize the probability of *bit error.* This can be done by using the criterion

$$\text{argmax}_{\tilde{x}_n} \Pr\{\tilde{x}_n|y_0,\ldots,y_{N-1}\}.$$

We will now see that the BCJR algorithm, a close relative of the Viterbi algorithm, can be used to implement such a detector efficiently.

We start by rewriting the decision criterion in a more convenient form. Assume again that the prior has product form, i.e., that

$$\Pr(\tilde{x}_0,\ldots,\tilde{x}_{N-1}) = \prod_{n=0}^{N-1} \Pr(\tilde{x}_n),$$

Then we get

$$\begin{aligned}
& \operatorname{argmax}_{\tilde{x}_n} \Pr\{\tilde{x}_n | y_0,\ldots,y_{N-1}\} \\
= \; & \operatorname{argmax}_{\tilde{x}_n} \sum_{\tilde{x}_0,\ldots,\tilde{x}_{n-1},\tilde{x}_{n+1},\ldots,\tilde{x}_{N-1}} \Pr\{\tilde{x}_0,\ldots,\tilde{x}_{N-1} | y_0,\ldots,y_{N-1}\} \\
= \; & \operatorname{argmax}_{\tilde{x}_n} \sum_{\tilde{x}_0,\ldots,\tilde{x}_{n-1},\tilde{x}_{n+1},\ldots,\tilde{x}_{N-1}} \Pr\{y_0,\ldots,y_{N-1} | \tilde{x}_0,\ldots,\tilde{x}_{N-1}\} \Pr\{\tilde{x}_0,\ldots,\tilde{x}_{N-1}\} \\
= \; & \operatorname{argmax}_{\tilde{x}_n} \sum_{\tilde{x}_0,\ldots,\tilde{x}_{n-1},\tilde{x}_{n+1},\ldots,\tilde{x}_{N-1}} \prod_{n=0}^{N-1} p(z_n = y_n - h_0\tilde{x}_n - \sum_{l=1}^{L-1} h_l \tilde{x}_{n-l}) \Pr(\tilde{x}_n) \\
= \; & \operatorname{argmax}_{\tilde{x}_n} \sum_{\tilde{x}_0,\ldots,\tilde{x}_{n-1},\tilde{x}_{n+1},\ldots,\tilde{x}_{N-1}} \prod_{n=0}^{N-1} e^{m(y_n;\tilde{x}_n;\sigma_n)+\ln \Pr(\tilde{x}_n)}
\end{aligned}$$

Consider now the trellis and assume that we label the edges of this trellis with the partial metrics $e^{m(y_n;\tilde{x}_n;\sigma_n)+\ln \Pr(\tilde{x}_n)}$ as for the Viterbi algorithm. The metric associated to each path through this trellis is then simply the product of the associated partial metrics traversed by this path. For the Viterbi algorithm we had to find the path with the largest such metric. If we consider the above equation then we see that for the problem of finding the most probable bit value we have to proceed as follows: Sum up the path metrics of all paths which have a specific value $\tilde{x}_n$ for the n-th transmitted symbol and decide upon the n-th transmitted symbol based on this sum. The BCJR algorithm is an efficient algorithm to accomplish this summation.

Consider all paths in the trellis which pass through a particular state at time i. For the Viterbi algorithm we realized that the path with the largest metric among these paths is the one which has the largest "past" metric and the largest "future" metric. This was true since the set of all such paths has a *product structure*, i.e., we can combine any "past" with any "future." Assume now that rather than determining the path with the largest metric we want to determine the sum of the metrics of *all* such paths. Again, because of the product structure, the sum of all metrics is equal to the sum of all "past" metrics multiplied with the sum of all "future" metrics. This follows from the simple distributive law

$$\sum_{i,j} a_i b_j = \left(\sum_i a_i\right)\left(\sum_j b_j\right).$$

This is the basis of the BCJR algorithm.

3. EQUALIZATION

Let's start again from the equivalent discrete time channel

$$y_n = \sum_k \mathcal{R}_g(k) x_{n-k} + z_n,$$

where z_n is a complex valued circularly symmetric Gaussian process with $\mathcal{R}_z(k) = N_0 \mathcal{R}_g(k)$. We have seen in the previous sections various methods of implementing an *optimal* receiver for this setup where the criterion of optimality was either to minimize the probability of sequence error or the probability of bit error. In all our previous discussions we assumed that there exists a sufficiently small integer L such that $\mathcal{R}_g(k) = 0$ for $k \geq L$, so that the complexity of the optimal receiver, which was proportional to $|\mathcal{X}|^{L-1}$, was sufficiently small.

In this section we will discuss receiver structures which are *suboptimal* but have a *lower complexity*. The basic idea underlying these receiver structures is to *filter* the sequence y_n in such a way as to either *eliminate* or at least *mitigate* the effect of the inter-symbol interference. We will see that in return such a filter *boosts* the variance of the noise. There is therefore an inherent tradeoff between eliminating ISI and keeping the noise variance low. When we discuss equalizers we make use of some basic facts of linear prediction. We will therefore start with a small discussion on linear prediction summarized in Appendix A.

3.1 DECISION FEEDBACK EQUALIZERS

Let's now return to our equivalent discrete time channel model

$$y_n = \sum_k \mathcal{R}_g(k) x_{n-k} + z_n, \tag{2.6}$$

where z_n is a complex valued circularly symmetric Gaussian process with $\mathcal{R}_z(k) = N_0 \mathcal{R}_g(k)$.

Consider the decision feedback equalizer shown in Fig. 2.3. It consists of a

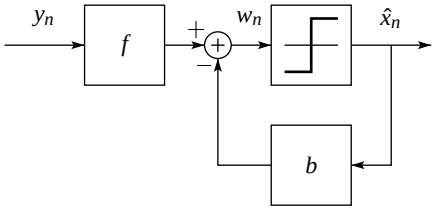


Figure 2.3: Block diagram of a decision feedback equalizer.

forward filter f, a *backward filter* b and a *decision device*. The idea is to feed

back the past decisions and to use them to cancel part of the intersymbol interference. Note that, under the assumption of correct past decisions, this feedback is *noiseless*! In order for the backward filter to be realizable we assume that it is *strictly causal*. It will be convenient to define the related monic and causal filter $B_0(z) = 1 + B(z)$.

In terms of the z-transform the input to the decision device, call it $W(z)$, is given by

$$W(z) = F(z)Y(z) - B(z)\hat{X}(z) = F(z)Y(z) - (B_0(z) - 1)\hat{X}(z).$$

There is no know *exact* analysis of this non-linear device. But we can gain some insight into the behavior of this device by assuming that the decision at the output of the decision device are *correct*, i.e., $\hat{x}_n = x_n$! Note also, that the most important performance measure is the probability of error at the output of the decision device. Unfortunately this quantity is very difficult to determine. We will therefore be content to determine the variance of the "noise" at the input of the decision device. The justification is that this noise variance is somewhat correlated to the probability of error (small/large variance implies small/large probability of error).

3.2 Minimum Mean Squared Error Criterion

In this section we will see how to choose the filters in such a way as to minimize the variance of "noise" at the input to the decision device.

Under the assumption of correct past decisions we have

$$\begin{aligned} E(z) &= W(z) - X(z) \\ &= (F(z)Y(z) - (B_0(z) - 1)X(z)) - X(z) \\ &= F(z)Y(z) - B_0(z)X(z) \\ &= F(z)Y(z) - V(z), \end{aligned}$$

where we defined $V(z) := B_0(z)X(z)$, $v_n := \sum_{k\geq 0} b_{0k}x_{n-k}$. Recall that we want to choose the filters f and b in such a way that the variance of the error sequence is minimized. We will proceed in two steps. Assume first that the backward filter b is fixed and that we want to choose the forward filter f in such a way as to minimize the variance of the error sequence. In the language of our previous discussion: we want to predict v_n based upon our the observations $\{y_k\}_{k\in\mathbb{Z}}$. Note in particular that we allow f to be non-causal. From Example 27 in Appendix A we know that the optimal filter is given by

$$F_{\text{MMSE}}(z) = \frac{\mathcal{S}_{v,y}(z)}{\mathcal{S}_y(z)}.$$

It remains to explicitly determine $\mathcal{S}_y(z)$ and $\mathcal{S}_{v,y}(z)$. Using (2.6), we see that

$$\begin{aligned}\mathcal{R}_y(k) &:= \mathbb{E}[y_n y_{n-k}^*] \\ &= \mathbb{E}\left[\left(\sum_m \mathcal{R}_g(m)x_{n-m} + z_n\right)\left(\sum_l \mathcal{R}_g^*(l)x_{n-k-l}^* + z_{n-k}^*\right)\right] \\ &= \sum_m \mathcal{R}_g(m)\mathcal{R}_g^*(m-k) + N_0\mathcal{R}_g(k) \\ &= \sum_m \mathcal{R}_g(m)\mathcal{R}_g(k-m) + N_0\mathcal{R}_g(k),\end{aligned}$$

and

$$\begin{aligned}\mathcal{R}_{v,y}(k) &:= \mathbb{E}[v_n y_{n-k}^*] \\ &= \mathbb{E}\left[\left(\sum_{m\geq 0} b_{0m}x_{n-m}\right)\left(\sum_l \mathcal{R}_g^*(l)x_{n-k-l}^* + z_{n-k}^*\right)\right] \\ &= \sum_{m\geq 0} b_{0m}\mathcal{R}_g^*(m-k) \\ &= \sum_{m\geq 0} b_{0m}\mathcal{R}_g(k-m).\end{aligned}$$

In the spectral domain this translates to

$$\mathcal{S}_y(z) = \mathcal{S}_g(z)(\mathcal{S}_g(z) + N_0), \quad \mathcal{S}_{v,y}(z) = B_0(z)\mathcal{S}_g(z).$$

Therefore

$$F_{\text{MMSE}}(z) = \frac{\mathcal{S}_{v,y}(z)}{\mathcal{S}_y(z)} = \frac{B_0(z)\mathcal{S}_g(z)}{\mathcal{S}_g^2(z) + N_0\mathcal{S}_g(z)} = \frac{B_0(z)}{\mathcal{S}_g(z) + N_0}.$$

We see that we expressed the optimal forward filter as a function of the backward filter. With this choice of forward filter the error sequence is given by

$$E(z) = F(z)Y(z) - B_0(z)X(z) = B_0(z)\left[\frac{Y(z)}{\mathcal{S}_g(z) + N_0} - X(z)\right] = B_0(z)U(z),$$

where we defined

$$U(z) := \frac{Y(z)}{\mathcal{S}_g(z) + N_0} - X(z). \tag{2.7}$$

What is the spectrum of the WSS process u_n? Note that if a_n and b_n are two WSS processes and $c_n = a_n - b_n$, then

$$\mathcal{S}_c(z) = \mathcal{S}_a(z) + \mathcal{S}_b(z) - \mathcal{S}_{a,b}(z) - \mathcal{S}_{a,b}^*(1/z^*).$$

We apply this fact to the process u_n, where from equation (2.7) we have $U(z) := \frac{Y(z)}{\mathcal{S}_g(z)+N_0} - X(z)$. The first part is the result of passing $Y(z)$ with spectrum $\mathcal{S}_y(z) =$

$\mathcal{S}_g(z)(\mathcal{S}_g(z)+N_0)$ through a filter $\frac{1}{\mathcal{S}_g(z)+N_0}$. Therefore the first part has a spectrum equal to

$$\mathcal{S}_g(z)(\mathcal{S}_g(z)+N_0)\frac{1}{\mathcal{S}_g(z)+N_0}\frac{1}{\mathcal{S}_g^*(1/z^*)+N_0}=\frac{\mathcal{S}_g(z)}{\mathcal{S}_g^*(1/z^*)+N_0}=\frac{\mathcal{S}_g(z)}{\mathcal{S}_g(z)+N_0}.$$

The second part has a spectrum equal to 1. It remains to determine the cross-correlation parts. If a_n and b_n are two WSS processes where $a_n=\sum_l g_l b_{n-l}$ then

$$\mathcal{R}_{a,b}(k)=\mathbb{E}[(\sum_l g_l b_{n-l})b_{n-k}^*]=\sum_l g_l \mathcal{R}_b(k-l)\Rightarrow \mathcal{S}_{a,b}(z)=G(z)\mathcal{S}_b(z).$$

It follows that the spectrum corresponding to the cross-correlation is equal to

$$\mathcal{S}_g(z)\frac{1}{\mathcal{S}_g(z)+N_0}\mathcal{S}_x(z)=\frac{\mathcal{S}_g(z)}{\mathcal{S}_g(z)+N_0}.$$

Combining all these results we get

$$\begin{aligned}\mathcal{S}_u(z) &= \frac{\mathcal{S}_g(z)}{\mathcal{S}_g(z)+N_0}+1-\frac{\mathcal{S}_g(z)}{\mathcal{S}_g(z)+N_0}-\frac{\mathcal{S}_g^*(1/z^*)}{\mathcal{S}_g^*(1/z^*)+N_0}\\ &= \frac{\mathcal{S}_g(z)}{\mathcal{S}_g(z)+N_0}+1-2\frac{\mathcal{S}_g(z)}{\mathcal{S}_g(z)+N_0}\\ &= \frac{N_0}{\mathcal{S}_g(z)+N_0}\end{aligned}$$

Example 8. [Linear Minimum Mean-Squared Equalizer] We are now ready to discuss the first important special case. This is the case where we *omit* the feedback filter and our equalizer consists solely of the forward filter f. This is called a *linear* equalizer and since we chose as a criterion for the filter to minimize the mean-squared error this is abbreviated as the LMMSE equalizer. In this case we have $B(z)=0$ or, equivalently, $B_0(z)=1$. We see from $E(z)=B_0(z)U(z)=U(z)$ that in this case the power spectrum of the error sequence is equal to $\mathcal{S}_u(z)=\frac{N_0}{\mathcal{S}_g(z)+N_0}$. Therefore the mean squared error incurred in this case is equal to

$$\epsilon^2_{\text{LE-MMSE}}=\int_{-\frac{1}{2}}^{\frac{1}{2}}\frac{N_0}{\mathcal{S}_g(e^{2\pi jf})+N_0}df. \tag{2.8}$$

□

In the general case where we allow a feedback filter we still have the degree of freedom in choosing b. Recall that

$$E(z)=B_0(z)U(z)=(B(z)+1)U(z)=B(z)U(z)+U(z),$$

where $B(z)$ is strictly causal. It follows from Example 26 in Appendix A that the optimal filter $B_0(z)$ in the mean-squared sense is the monic and causal whitening filter.

Assuming that $\mathcal{S}_g(z)$ is rational then $\mathcal{S}_u(z)$ is rational. In this case we have seen how to derive the whitening filter. Under suitable conditions whitening filters can also be constructed for non-rational spectra. This is discussed in Appendix B.

From Appendix B we know that under suitable conditions we can factor $\mathcal{S}_u(z)$ as

$$\mathcal{S}_u(z) = \mathcal{S}_u^+(z)\mathcal{S}_u^-(z),$$

where $\mathcal{S}_u^+(z)$ is causal and $\mathcal{S}_u^-(z)$ is anticausal and that

$$\frac{\mathcal{S}_u^{\pm}(z)}{\sqrt{\exp\left\{\int_{-\frac{1}{2}}^{\frac{1}{2}} \ln[\mathcal{S}_u(e^{2\pi jf})]\,df\right\}}},$$

is a monic and causal/anticausal filter. Our monic and causal whitening filter is therefore

$$B_0(z) := \frac{\sqrt{\exp\left\{\int_{-\frac{1}{2}}^{\frac{1}{2}} \ln[\mathcal{S}_u(e^{2\pi jf})]\,df\right\}}}{\mathcal{S}_u^+(z)}.$$

With this choice $\mathcal{S}_e(z)$ is equal to

$$\begin{aligned}
\mathcal{S}_e(z) &= \mathcal{S}_u(z)B_0(z)B_0^*(1/z^*) \\
&= \frac{\mathcal{S}_u(z)}{\mathcal{S}_u^+(z)\left(\mathcal{S}_u^+(1/z^*)\right)^*}\exp\left\{\int_{-\frac{1}{2}}^{\frac{1}{2}} \ln\left[\mathcal{S}_u(e^{2\pi jf})\right]df\right\} \\
&= \frac{\mathcal{S}_u(z)}{\mathcal{S}_u^+(z)\mathcal{S}_u^-(z)}\exp\left\{\int_{-\frac{1}{2}}^{\frac{1}{2}} \ln\left[\mathcal{S}_u(e^{2\pi jf})\right]df\right\} \\
&= \exp\left\{\int_{-\frac{1}{2}}^{\frac{1}{2}} \ln\left[\mathcal{S}_u(e^{2\pi jf})\right]df\right\} \\
&= \exp\left\{\int_{-\frac{1}{2}}^{\frac{1}{2}} \ln\left[\frac{N_0}{\mathcal{S}_g(e^{2\pi jf})+N_0}\right]df\right\}.
\end{aligned}$$

The noise variance in this case is therefore

$$\epsilon^2_{\text{DFE-MMSE}} = \int_{-\frac{1}{2}}^{\frac{1}{2}} \mathcal{S}_e(e^{2\pi jf})df = \exp\left\{\int_{-\frac{1}{2}}^{\frac{1}{2}} \ln\left[\frac{N_0}{\mathcal{S}_g(e^{2\pi jf})+N_0}\right]df\right\}. \quad (2.9)$$

It is constructive to compare $\epsilon^2_{\text{DFE-MMSE}}$ with $\epsilon^2_{\text{LE-MMSE}}$. Recall that $\ln(x)$ is a concave function and that therefore by Jensen's inequality

$$\int \ln f \leq \ln \int f,$$

provided the two integrals exist. Applying this fact to (2.8) and (2.9) we see immediately that

$$\epsilon^2_{\text{DFE-MMSE}} \leq \epsilon^2_{\text{LE-MMSE}},$$

as was to be expected.

3.3 ZERO FORCING CRITERION

We can also choose our filters according to a different criterion, namely in such a way that we completely remove the ISI. For the LE this choice is investigated in Exercise 2.10, where it is shown that

$$\epsilon^2_{\text{LE-ZF}} = N_0 \int_{-\frac{1}{2}}^{\frac{1}{2}} \frac{1}{\mathcal{S}_g(e^{2\pi jf})} df.$$

For the more general DFE we will undertake this investigation now.

Assume we choose

$$\begin{aligned} F(z) &= \frac{1}{\sqrt{\exp\left\{\int_{-\frac{1}{2}}^{\frac{1}{2}} \ln[\mathcal{S}_g(e^{2\pi jf})]\, df\right\}} \mathcal{S}_g^-(z)} \\ B_0(z) &= \frac{\mathcal{S}_g^+(z)}{\sqrt{\exp\left\{\int_{-\frac{1}{2}}^{\frac{1}{2}} \ln[\mathcal{S}_g(e^{2\pi jf})]\, df\right\}}} \end{aligned}$$

With this choice we have

$$\begin{aligned} E(z) &= F(z)Y(z) - B_0(z)X(z) \\ &= F(z)\left(\mathcal{S}_g(z)X(z) + Z(z)\right) - B_0(z)X(z) \\ &= \left(F(z)\mathcal{S}_g(z) - B_0(z)\right)X(z) + F(z)Z(z) \\ &= \left(\frac{\mathcal{S}_g^+(z)\mathcal{S}_g^-(z)}{\sqrt{\exp\left\{\int_{-\frac{1}{2}}^{\frac{1}{2}} \ln[\mathcal{S}_g(e^{2\pi jf})]\, df\right\}} \mathcal{S}_g^-(z)} - \frac{\mathcal{S}_g^+(z)}{\sqrt{\exp\left\{\int_{-\frac{1}{2}}^{\frac{1}{2}} \ln[\mathcal{S}_g(e^{2\pi jf})]\, df\right\}}} \right) X(z) + F(z)Z(z) \\ &= F(z)Z(z) \\ &= \frac{Z(z)}{\sqrt{\exp\left\{\int_{-\frac{1}{2}}^{\frac{1}{2}} \ln[\mathcal{S}_g(e^{2\pi jf})]\, df\right\}} \mathcal{S}_g^-(z)}. \end{aligned}$$

We see that indeed all the ISI has been eliminated. Recall that $\mathcal{S}_z = N_0 \mathcal{S}_g(z)$. It follows that the spectrum of $E(z)$ is flat. More precisely,

$$E(z) = \frac{N_0}{\exp\left\{\int_{-\frac{1}{2}}^{\frac{1}{2}} \ln[\mathcal{S}_g(e^{2\pi jf})]\, df\right\}},$$

and so the associated mean squared error is equal to

$$\epsilon^2_{DFE-ZF} = \frac{N_0}{\exp\left\{\int_{-\frac{1}{2}}^{\frac{1}{2}} \ln[\mathcal{S}_g(e^{2\pi jf})]\, df\right\}} = N_0 \exp\left\{\int_{-\frac{1}{2}}^{\frac{1}{2}} \ln\left[\frac{1}{\mathcal{S}_g(e^{2\pi jf})}\right] df\right\}$$

Using again Jenson's inequality, we see that, as in the case when using the MMSE criterion, the DFE leads to a lower variance of the error sequence, i.e.,

$$\epsilon^2_{DFE-ZF} \leq \epsilon^2_{LE-ZF}.$$

For the DFE-ZF a *precoding technique* has been proposed to avoid the problem with unreliable feeback. The basic idea is to move the effect of the feedback filter into the transmitter part. This of course requires that the transmitter knows the channel. Assume that rather than transmitting $X(z)$ we transmit $\tilde{X}(z) := \frac{X(z)}{B_0(z)}$. The received sequence $\tilde{Y}(z)$ is then equal to

$$\tilde{Y}(z) = \frac{\mathcal{S}_g(z)}{B_0(z)} X(z) + Z(z),$$

where the noise is again circularly symmetric Gaussian with power spectral density equal to $N_0 \mathcal{S}_g(z)$. At the output of the filter $F(z)$ we then have

$$\tilde{Y}(z)F(z) = \frac{\mathcal{S}_g(z)F(z)}{B_0(z)} X(z) + F(z)Z(z) = X(z) + F(z)Z(z).$$

We see that the ISI has been removed and that the noise is white! In effect we have cancelled the causal part of the ISI by the prefilter (which does not suffer from error propagation) and we cancel the anticausal part by the forward filter. Not all is well though. The problem with this naive implementation of precoding is that it boosts the *transmit power*!

A simple trick eliminates this problem almost completely. Assume that our constellation is $\mathcal{X} := \{\pm\Delta, \pm 3\Delta, \pm(M-1)\Delta\}$ and let the transmitted sequence be

$$\tilde{X}(z) := \frac{X(z) + 2M\Delta\Xi(z)}{B_0(z)},$$

where ξ_n is an integer sequence chosen in such a way that $\tilde{x}_n$ is in the range $(M\Delta, M\Delta]$ (this can be done by simply taking the sequence and reducing it modulo $2M\Delta$ before transmission). Now the received sequence $\tilde{Y}(z)$ is equal to

$$\tilde{Y}(z) = \frac{\mathcal{S}_g(z)}{B_0(z)}(X(z) + 2M\Delta\Xi(z)) + Z(z),$$

and the output of the filter $F(z)$ is equal to

$$\begin{aligned} \tilde{Y}(z)F(z) &= \frac{\mathcal{S}_g(z)F(z)}{B_0(z)}(X(z) + 2M\Delta\Xi(z)) + F(z)Z(z) \\ &= X(z) + 2M\Delta\Xi(z) + F(z)Z(z). \end{aligned}$$

Since the noise is white we can make a decision on $X(z) + 2M\Delta\Xi(z)$ and a modulo $2M\Delta$ operation then gives us an estimate of $X(z)$.

How does this effect the transmit power? The original constellation had an average energy per transmitted symbol equal to

$$\frac{2}{M}\Delta^2\sum_{i=1}^{M/2}(2i-1)^2=\Delta^2\frac{M^2-1}{3}.$$

Assuming that the values of the sequence $\tilde{x}$ are uniformly distributed over the intervale $(-M\Delta,M\Delta]$ the average energy per symbol expanded is equal to

$$\frac{1}{M}\Delta^2\int_0^M a^2da=\Delta^2\frac{M^2}{3},$$

which is only insignificantly higher.

3.4 SUMMARY

We summarize: We investigated four (one of them in Exercise 2.10) equalizers and determined the resulting noise variance at the input to the decision device. The first two equalizers were *linear equalizers* (LE), i.e., they consists simply of a forward filter. This forward filter can be chosen either in such a way as to cancel completely the ISI (zero forcing (ZF) criterion) or in such a way as to minimize the mean squared error (MMSE criterion). The second two examples were decision feedback equalizers. Again, the filters can be chosen either according to the ZF criterion or according to the MMSE criterion. We showed that

$$\epsilon^2_{\text{DFE-MMSE}}\le\epsilon^2_{\text{LE-MMSE}}\le\epsilon^2_{\text{LE-ZF}}.$$

and

$$\epsilon^2_{\text{DFE-MMSE}}\le\epsilon^2_{\text{DFE-ZF}}\le\epsilon^2_{\text{LE-ZF}}.$$

In the sequel we list the choice of filters and the resulting noise variance for each of these three cases.

EXERCISES

2.1. [Trellis Sections] Assume again that we use antipodal signaling. In class we draw the trellis digram for the case $L=2$. Draw one trellis section for the cases $L=3$ and $L=4$. For the case of general L. What is the size of the state space and how many edges are there per trellis section?

2.2. Consider the following transmission scheme. The transmitted symbols x_n are i.i.d. random variables, taking on $+1$ and -1 equally likely, and the received symbols are given by

$$y_n=\prod_{i=1}^{n}x_i+z_n,\ \ n=1,\cdots,N,$$

where z_n is an i.i.d. sequence of random variables with density $p(z)$.

We would like to use the Viterbi (BCJR) algorithm to find the most likely transmitted sequence (bit) given the received sequence. Find a suitable *state* (which has a small state space) so that we can write $p(y_1,\cdots,y_N|x_1,\cdots,x_N)$ in the form

$$p(y_1,\cdots,y_N|x_1,\cdots,x_N)=\prod_{n=1}^{N} f(y_n;x_n;\sigma_n).$$

Draw the corresponding trellis, assuming that $N=4$. What is the complexity of the decoding algorithm? NOTE: This part requires quite some calculations. Finish the other problems first and do not loose too much time on this part!

Now assume that

$$p(z):=\begin{cases}\frac{2-|z|}{4}, & |z|\le 2,\\ 0, & \text{otherwise.}\end{cases}$$

Assume further that

$$y_1=-0.1, y_2=0.5, y_3=0.9, y_4=-0.2.$$

Apply the Viterbi algorithm to this case to find the most likely transmitted sequence $\tilde{x}_1,\tilde{x}_2,\tilde{x}_3,\tilde{x}_4$.

2.3. [Viterbi Algorithm] Assume that we have the following discrete-time channel:

$$y_n=\sum_{l=0}^{L-1} h_l x_{n-l}+z_n,\ \ n=0,\ldots,N-1,$$

where all quantities are complex-valued, x_n takes values in some discrete subset of $\mathbb{C}$, h represents the channel impulse response which is causal and of length L and the z_n are i.i.d. and circularly-symmetric Gaussian. We are interested in determining the likelihood of the most likely sequence, i.e., we want to determine

$$\max_{\tilde{x}_0,\ldots,\tilde{x}_{N-1}} p(y_0,\ldots,y_{N-1}|\tilde{x}_0,\ldots,\tilde{x}_{N-1}).$$

Devise an efficient algorithm to accomplish this task.

2.4. Consider transmission over a linear time-invariant channel using pulse-amplitude modulation as discussed in class, i.e., the received signal is of the form

$$y_E(t)=\sum_{n=0}^{N-1} x_n g_E(t-nT)+Z(t),$$

where the symbols x_n take values in the symbol alphabet $\mathcal{X}$ and where $Z(t)$ is assumed to be white circularly-symmetric Gaussian noise. Assume, as we did in class, that for some finite natural number L we have

$$\mathcal{R}_g(k)=0,\ \ |k|\ge L,$$

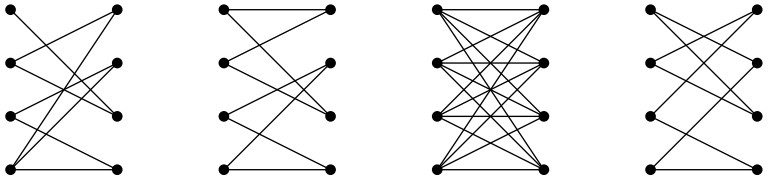
where

$$\mathcal{R}_g(k) := \int g_E(t) g_E^*(t - kT) dt.$$

Recall that the optimum detector for this case can be implemented efficiently by means of the *Viterbi* algorithm.

1. Which of the following pictures could represent a trellis section for this problem.

2. For those pictures that could represent a trellis section for this problem, what are the corresponding parameters $|\mathcal{X}|$ and L?



(a) (b) (c) (d)

2.5. [Preliminaries for the BCJR Algorithm] Consider the same transmission model as in the previous example. The Viterbi algorithm finds the most likely *sequence* and, under the assumption of a uniform prior on the set of all such sequences, minimizes the *sequence error probability*. Sometimes we are more interested in minimizing the *bit error probability*, which we can accomplish if we use the decision criterion

$$\max_{\tilde{x}_n} \Pr\{\tilde{x}_n | y_0, \ldots, y_{N-1}\}.$$

In class we will get to know the BCJR algorithm (which is a close relative of the Viterbi algorithm) to accomplish this task efficiently. As a preparation: Express $\Pr\{\tilde{x}_n | y_0, \ldots, y_{N-1}\}$ in terms of quantities which we can compute (efficiently or not).

2.6. [Shortest Path Algorithm] The Viterbi algorithm is a special case of a more general principle called *dynamic programming*. In this example we will investigate another application of dynamic programming. Assume we want to find the shortest route between a given pair of cities in Europe. We are given a map which contains the cities and the streets between them, labeled with the length of these streets. We envision this map as a *graph*: the cities are *nodes* and the streets are *edges*; each edge $\mathtt{e}$ has an associated *length* $l(\mathtt{e})$, $l(\mathtt{e}) > 0$, which corresponds to the length of the corresponding street. Given a pair of nodes $(\mathtt{v}_S, \mathtt{v}_E)$, a *path* is a sequence of connected edges starting at $\mathtt{v}_S$ and ending at $\mathtt{v}_E$. Given a pair of nodes $(\mathtt{v}_S, \mathtt{v}_E)$, and we are asking for the *shortest* path between them, where the length of a *path*, is the sum of the lengths of the traversed edges. We will now describe an algorithm, very similar in flavor to the Viterbi algorithm, which solves this problem efficiently. The algorithm is due to Dijkstra (1959).

Read the handouts concerning this topic and apply the algorithm to the example given therein.

2.7. [Partial Sequence Estimator] Consider again antipodal signaling over the discrete-time channel

$$y_n = \sum_{l=0}^{L-1} h_l x_{n-l} + z_n, \;\; n = 0, \ldots, N-1,$$

where $\{h_l\}_l$ represents the channel impulse response which is causal and of length $L \geq 3$ and the z_n are i.i.d. Gaussian distributed. The *Viterbi* algorithm is a sequence APP estimator, the *BCJR* algorithm is a symbol APP estimator. We are now interested in an algorithm maximizing the probability of two consecutive symbols (x_{i-1}, x_i). Derive such an algorithm.

2.8. In this problem we will develop some basic properties of minimum phase systems step by step. Recall that a rational filter $H(z)$ is called minimum phase if all its poles and zeros are inside the unit circle.

1. Show that a minimum phase filter $H(z)$ is causal and stable.
2. Show that a minimum phase filter $H(z)$ has a causal and stable *inverse*.
3. Show that the frequency response of a filter $H(z)$ of the form

   $$H(z) := \frac{z^{-1} - a^*}{1 - az^{-1}}$$

   has unit magnitude, i.e., $|H(e^{2\pi jf})| = 1$. Such a filter is called an *all-pass* filter, since it passes all frequency components with a gain of unity.
4. Assume that $H(z)$ is a rational filter with all its poles inside the unit circle and assume further that also all its zeros are inside the unit circle except the zero at $z = \frac{1}{c^*}$, where $|c| < 1$, i.e.,

   $$H(z) = H_1(z)(z^{-1} - c^*),$$

   where $H_1(z)$ is a minimum phase filter (i.e., has all its poles *and* zeros inside the unit circle). Show how we can derive from $H(z)$ a minimum phase filter with equal frequency response. Generalize this approach to the case where $H(z)$ has any number of zeros outside the unit circle.
5. As stated in class, a rational filter $H(z)$ which is minimum phase has the following important *minimum energy-delay* property. Of all rational filters $H(z)$ with the same $|H(e^{2\pi jf})|^2$, a minimum phase filter *maximizes* the partial energy terms

   $$\sum_{i=0}^{k} |h_i|^2 \tag{2.10}$$

for all $k \geq 0$. Note that in the limit though we get from Parseval that

$$\sum_{i=0}^{\infty} |h_i|^2 = \int_{-\frac{1}{2}}^{\frac{1}{2}} |H(e^{2\pi jf})|^2 df,$$

which only depends on the magnitude of the frequency response and is therefore equal for all filters with the same such magnitude. In this exercise we will prove the assertion (2.10) for $k = 0$.

Note from the previous item that we can write any rational filter $H(z)$ as the product of a minimum phase filter with an all-pass filter, i.e.,

$$H(z) = H_{\min}(z)H_{\text{ap}}(z).$$

Recall the *initial value* theorem of the z-Transform. For a *causal* h_n we have $h_0 = \lim_{z\to\infty} H(z)$. Use these two facts together to prove the assertion for $k = 0$.

2.9. Consider the setup in Example 26. Prove that the defining equations in (A.4) indeed imply that f should be the monic and causal whitening filter by completing the following steps.

1. Let x_n be a zero mean WSS stochastic process and $w_n = \sum_{k\geq 0} f_k x_{n-k}$ be the result of passing x_n through a causal and monic filter f_n.
2. Let $\mathcal{R}_x(k) := \mathbb{E}[x_n x^*_{n-k}]$. Show that
$$\mathcal{R}_w(k) = \sum_{l\geq k} f^*_{l-k} \sum_{m\geq 0} f_m \mathcal{R}_x(l-m).$$
3. Assume first that f fulfils the equations (A.4). Show that this implies that for $k \geq 1$, $\mathcal{R}_w(k) = 0$.
4. Now argue that this implies that $\mathcal{R}_w(k) = 0$ for $k \neq 0$. This shows that w is white, i.e., that f is indeed a whitening filter (it is monic and causal by definition).

2.10. In this exercise we will investigate the so called linear zero forcing equalizer. Consider again the equivalent discrete time channel model

$$y_n = \sum_k \mathcal{R}_g(k)x_{n-k} + z_n,$$

where z_n is a complex valued circularly symmetric Gaussian process with $\mathcal{R}_z(k) = N_0\mathcal{R}_g(k)$. Assume we filter this received signal through some filter $F(z)$.

1. How do we have to choose f in order to eliminate the intersymbol interference completely. (This is the reason why this design criterion is called zero forcing since we force the intersymbol interference to zero.)

2. What is the power spectral density of the noise at the output of the filter.

3. Argue that the noise power for this design criterion is larger than for the MMSE criterion.

2.11. Show that

$$\epsilon^2_{\text{DFE-ZF}} \leq \epsilon^2_{\text{LE-ZF}}.$$

2.12. Consider the naive precoding scheme discussed in class where the transmitted signal is equal to

$$\frac{X(z)}{B_0(z)},$$

where $B_0(z) := \frac{S_g^+(z)}{A_g}$. Assume that $\mathcal{R}_x(k) = \delta(k)$. Show that $\mathcal{R}_{\tilde{x}}(0) \geq 1$.

3

SPREAD SPECTRUM COMMUNICATIONS

1. MULTIPLE ACCESS COMMUNICATIONS

So far we have been concerned with a single user transmitting over a dedicated channel. In a *multiple access* system there are several users who want to transmit information to a single *receiver*, see Fig. 3.1. Think e.g. of a cellular mobile

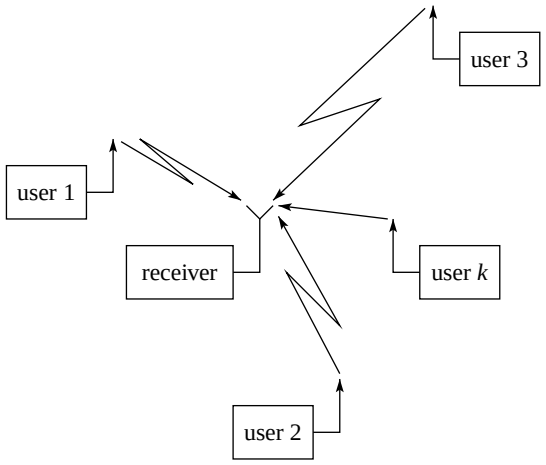


Figure 3.1: The basic multiple-access problem. Several users try to convey information to the same receiver.

phone system in which all users in the same cell want to communicate to (via) the same base station.

The multiple access problem can be solved in many different ways. One natural method is to use *time-division multiple access* (TDMA). Assume that there are k users who want to convey information to the same receiver. Split the time axis into many (small) slots and assign each such slot to exactly one user. Each user only transmits at her assigned time slots, so that at any point in time only a single user is transmitting. This reduces the multiple access problem to k independent point-to-point channels. For each such point-to-point problem we can use the standard methods we have learned so far. Note that in order for this scheme to work we need all users to be synchronized and that we have to consider the time delays incurred by the channel. In a similar way one can split the *frequency band* into k disjoint frequency bands and assign one such band to each user who limits his transmission to his assigned frequency band. Such a scheme is called *frequency-division multiple access.* Again, the multiple access problem is reduced to several independent point-to-point communication problems. The basic idea of the above two methods is to *separate* the transmission of the individual users by assigning them subspaces which are *orthogonal*. For TDMA this orthogonality is most easily seen in the time domain, whereas for FDMA this orthogonality is easier to see in the frequency domain. More generally, by assigning to individual users orthogonal subspaces such a separation can be achieved and, in general, such a method is called *code-division multiple access* (CDMA). To see another example, assume that we use a Nyquist pulse $\psi(t)$, so that $\psi(t)$ is orthogonal to all its shifts $\psi(t - i\tau)$. Assigning each such shift to exactly one user and assume that each user only employs the subspace which is spanned by the set of shifts assigned to him. Then we can be assured that the transmissions of individual users are orthogonal.

TDMA, FDMA and CDMA, although they constitute quite natural approaches, are, in general, not optimal, i.e., for a given channel model (frequency band, noise) and a given transmit power, they, in general, fall short of achieving the maximal information throughput possible. In the information theory class you will learn how to assess the performance of these schemes in an information theoretic sense and how this performance compares to the ultimate limits.

Although TDMA, FDMA and CDMA are not necessarily optimal, they are the preferred multiple-access schemes in practice since they allow the transmission of a sizeable fraction of capacity at fairly low complexity. In this section we will learn some basic facts about *spread spectrum* communications, a transmission technique which can be used for point-to-point channels as well as a multiple access technique.

2. SPREAD SPECTRUM

Spread spectrum techniques can be used for point-to-point channels as well as a multiple access technique. Consider first the point-to-point case. The idea is to *spread* out the signal of the given user over a frequency band which is much larger

than required and to use a signal which in the frequency domain looks like "white noise" over the frequency band used.

The basic transmitter/receiver block diagram of a spread spectrum system for a single user is shown in Fig. 3.2. Here, for simplicity, we assume that antipo-

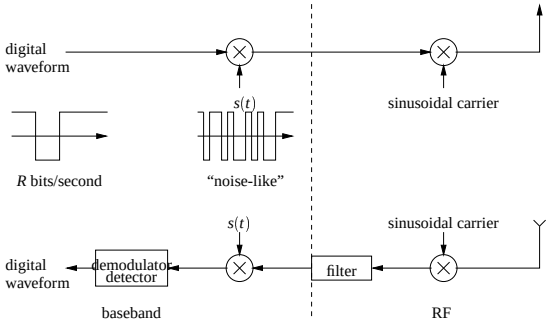


Figure 3.2: Basic transmitter/receiver block diagram of a spread spectrum system.

dal signaling is used and that the transmit filter is simply a rectangular. Before modulating the waveform into passband the waveform is *spread* by a wideband "noise-like" *signature* $s(t)$.

3. SPREAD SPECTRUM MULTIPLE ACCESS

In a multiple-access scenario the system looks as shown in Fig. 3.3. Note that

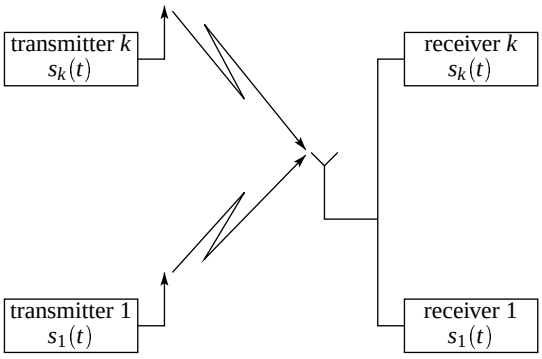


Figure 3.3: A spread spectrum system with k users.

each user is assigned a different signature. These signatures are chosen to be

almost orthogonal.[1]

4. A First (Very Shaky) Analysis

4.1 Analysis Of Corresponding Narrowband System

Consider first the corresponding narrowband system shown in Fig. 3.4. Assume

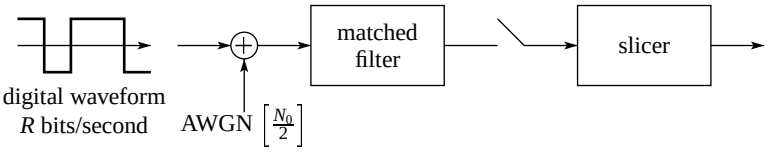


Figure 3.4: The corresponding narrowband system.

we use antipodal signaling and a rectangular transmit filter. If we assume that we use an energy of E_b per bit then the signal constellation viewed in signal space is the one given in Fig. 3.5. Recall that the AWGN has variance $\sigma^2 = \frac{N_0}{2}$. It follows

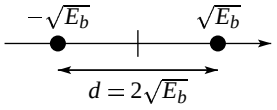


Figure 3.5: Antipodal signal constellation.

that the corresponding bit error probability is equal to

$$P_b = Q\left(\frac{d}{2\sigma}\right) = Q\left(\sqrt{\frac{2E_b}{N_0}}\right).$$

This is for an uncoded system, but even in the presence of coding the bit error probability is a function of $\frac{E_b}{N_0}$.

4.2 Analysis of Spread Spectrum Multiple Access System

Consider the receiver for the j-th user of a spread spectrum multiple access (SSMA) system with k users as shown in Fig. 3.6. We will assume that the *background* noise is negligible compared to the *interference* from other users. We will assume

[1] Actually we will see that these signatures can be chosen such that any pair of such signatures is almost orthogonal even if we allow arbitrary time shifts. This is important if we allow *asynchronous* users.

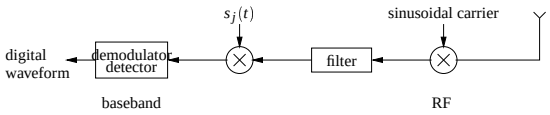


Figure 3.6: Receiver for the j-th out of k users.

for now that the interference from other users can be modeled as white Gaussian noise so that we will only have to be concerned with the variance of this noise.

Assume that in order to achieve the desired bit probability of error we require an $\frac{E_b}{N_0}$ of $\left(\frac{E_b}{N_0}\right)_{\text{req}}$. This number depends on the specific modulation scheme (in our example antipodal) and the coding scheme used. Assume that the received power of each user is the same and is equal to P_S. Since there are $(k-1)$ *other* users the total received power of all other users is equal to $I_0 := P_S(k-1)[W]$. By assumption this power will act as AWGN noise. This power is *spread* over a bandwidth of $W[Hz]$, so that the noise power spectral density of this interference is equal to $\frac{P_S(k-1)}{W}$, call it I_0. I_0 plays the role equivalent to N_0. The energy per (transmitted information) bit is equal to

$$E_b = \frac{P_S}{R}.$$

Therefore,

$$\frac{E_b}{I_0} = \frac{P_S}{R}\frac{W}{P_S(k-1)} = \frac{W}{R(k-1)}.$$

It follows that in order to achieve the desired P_e we need

$$\frac{W}{R(k-1)} = \frac{E_b}{I_0} \geq \left(\frac{E_b}{N_0}\right)_{\text{req}},$$

or

$$k-1 \leq \frac{W/R}{\left(\frac{E_b}{N_0}\right)_{\text{req}}}.$$

There are other important factors which influence the performance of a SSMA system. If the system is used for voice traffic then typically users are only speaking for a fraction of the time, i.e., only a fraction of the users is typically active at the same time. This is modeled by the *voice duty factor* G_V, $G_V > 1$. Instead of using I_0 we then use I_0/G_V. Another gain comes from the antenna. Often antennas are directional, i.e, there are maybe three antennas per cell, each covering an angle of 120 degrees. For this particular case, the antenna which receives the transmission of user j typically will only "see" a third of all other users as interference. We denote this gain by G_A.

On the other hand we will get interference from other cells. This is called *intercell interference* and is usually modelled as a factor $(1+f)$. Taking this factors into account we have the modified equation

$$k-1 \leq \frac{W/R}{\left(\frac{E_b}{N_0}\right)_{\text{req}}} \frac{G_A G_V}{1+f}$$

In the following lectures we will try to give a more reasonable analysis of such a system and to describe some of the necessary elements in more detail.

5. PSEUDORANDOM SEQUENCES

In the previous pages we have seen that one crucial element of a spread spectrum system is the generation of "noise-like" sequences. Ideally, we would like to employ spreading sequences $s(t)$ which are the realizations of AWGN processes with a bandwidth equal to the bandwidth used. But there are practical concerns. One has to be able to generate these signature waveforms with reasonable hardware complexity and, further, the same signature must be available at the transmitter and the receiver. For this reason one usually employs *binary pseudorandom sequences* generated by *linear feedback shift registers* (LFSR). These sequences have many of the properties that one would expect from truly random sequences and are very easy to implement in hardware.

5.1 MAXIMAL LENGTH LINEAR FEEDBACK SHIFT REGISTERS

We are interested in generating "random-like" binary sequences, i.e, sequences of elements from $\{0,1\}$ which resemble as much as possible sequences of fair coin flips. In particular we will focus on the following properties of sequences of fair coin flips.

- R.1 The relative frequencies of 0 and 1 are one-half.
- R.2 The probability of a run (of zeros or ones) of length k is equal to 2^{-k}.
- R.3 The correlation between pairs of bit positions which are a fixed non-zero constant apart is equal to one quarter.

We would like our sequences to fulfil the above properties as closely as possible.

Here we are only interested in *binary sequences*, i.e., sequences which take components in $\{0,1\}$. Recall that if we have two binary elements, call them a and b, then we can *add* them

$$a+b := \begin{cases} 0, & a=b, \\ 1, & a \neq b, \end{cases}$$

which corresponds to the logical XOR and we can *multiply* them

$$a \cdot b := \begin{cases} 1, & a = b = 1, \\ 0, & \text{else}, \end{cases}$$

which corresponds to the logical AND. Consider the binary recursion

$$s_n = \frac{1}{c_0} \sum_{i=1}^{r} s_{n-i} c_i, \;\; n \geq 0, \;\; \text{or, equivalently, } \sum_{i=0}^{r} s_{n-i} c_i = 0, \;\; n \geq 0, \tag{3.1}$$

where $s_{-r}, \cdots, s_{-1}$ are the *initial* values and where s_n, $n \geq -r$, and the *coefficients* c_i, $i \in \{0, \cdots, r\}$, are binary. Clearly, in order for the recursion (3.1) to be meaningful we need $c_0 = 1$, and in the sequel we will assume that this is the case. We can think of this sequence as being generated by the binary LFSR as shown in Fig. 3.7, where the initial values $s_{-r}, \cdots, s_{-1}$ are *preloaded*.

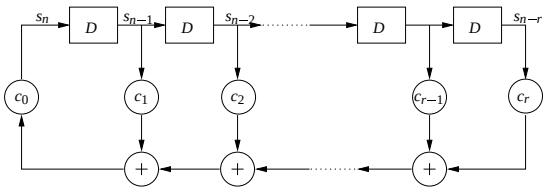


Figure 3.7: A binary linear feedback shift register of length r. We will always assume that $c_0 \neq 0 \neq c_r$.

It will turn out to be convenient to specify the sequence $s_0, s_1, \cdots$ by means of its *generating function* $G(D)$. Here, $G(D)$ is *defined* by

$$G(D) := \sum_{n=0}^{\infty} s_n D^n. \tag{3.2}$$

This is a *formal power sum*[2] in D and one should think of this formal power sum simply as a clothesline onto which the elements of the sequence $s_0, s_1, \ldots$ can be attached conveniently. In particular, for a formal power sum we are not concerned with issues of *convergence*.

[2]The name generating *function* is strictly speaking a misnomer since one should not think of $G(D)$ as a function but simply as a formal sum.

If we insert (3.1) into (3.2) we get

$$\begin{aligned} G(D) &= \sum_{n=0}^{\infty} s_n D^n \\ &\overset{(3.1)}{=} \sum_{n=0}^{\infty} \left(\sum_{i=1}^{r} s_{n-i} c_i \right) D^n \\ &= \sum_{i=1}^{r} c_i D^i \left(\sum_{n=0}^{\infty} s_{n-i} D^{n-i} \right) \\ &= \sum_{i=1}^{r} c_i D^i \left(\sum_{j=-i}^{-1} s_j D^j + G(D) \right) \\ &= \underbrace{\sum_{i=1}^{r} c_i D^i \left(\sum_{j=-i}^{-1} s_j D^j \right)}_{g_0(D)} + G(D) \sum_{i=1}^{r} c_i D^i \end{aligned}$$

If we define the binary polynomial $c(D) := \sum_{i=0}^{r} c_i D^i$ then we get the relation

$$G(D)c(D) = g_0(D),$$

where $g_0(D)$ is a *polynomial* of degree at most $(r-1)$ which depends on the initial values and the *feedback polynomial* $c(D)$, a polynomial of degree r. We conclude that

$$G(D) = \frac{g_0(D)}{c(D)}.$$

BASIC PROPERTIES OF LFSRS

We are now ready to investigate the basic properties of LFSRs.

Definition 6. [Periodicity] We say that a sequence $s_0, s_1, \ldots$ is *periodic* with period p if for all $i \geq 0$, $s_{i+p} = s_i$. We say that a sequence is *eventually periodic* if the above statement is true for all $i \geq i_0$, for some suitable constant i_0.

Lemma 1. [Property P.1] Consider a LFSR of memory r, $r \geq 1$, with feedback polynomial $c(D)$, $c_0 \neq 0$, and let $G(D)$ be the generating function of the sequence generated by the LFSR. Then $G(D)$ is eventually periodic with period p satisfying $p \leq 2^r - 1$.

Proof. Call the contents of the r shift registers at a given time i the *state* of the LFSR at time i. Note that the future of the evolution of the LFSR is completely specified once the state is known. Once the LFSR is in the all-zero state it will remain in this state forever and emit the all-zero sequence. Such a sequence is clearly periodic with period one and since $1 \leq 2^r - 1$ if $r \geq 1$, the claim is fulfilled

in this case. Further, there are only $2^r - 1$ non-zero states. This shows that also in the case that the LFSR never visits the zero state the output sequence is eventually periodic and that the period is at most $2^r - 1$. □

Lemma 2. Consider a LFSR of memory r, $r \geq 1$, with feedback polynomial $c(D)$, $c_0 \neq 0$, and let $G(D)$ be the generating function of the sequence generated by the LFSR. If $c_r \neq 0$ then $G(D)$ is periodic.

Proof. By Lemma 1 we know that $G(D)$ is eventually periodic. Call the period p. Let n_0, $n_0 \geq 0$, be the least integer n such that $s_i = s_{i+p}$ for all $i \geq n$. If $n_0 = 0$, then we are done. If on the contrary $n_0 > 0$ but $c_r = 1$ then note that from (3.1) we have

$$s_{n-r}c_r = \sum_{i=0}^{r-1} s_{n-i}c_i.$$

Therefore,

$$s_{n_0-1+p} = \frac{1}{c_r}\sum_{i=0}^{r-1} s_{n_0+(r-1)-i+p}c_i.$$

But also

$$s_{n_0-1} = \frac{1}{c_r}\sum_{i=0}^{r-1} s_{n_0+(r-1)-i}c_i = \frac{1}{c_r}\sum_{i=0}^{r-1} s_{n_0+(r-1)-i+p}c_i,$$

showing that $s_{n_0-1} = s_{n_0-1+p}$, a contradiction to the assumption that n_0 was the smallest such integer. □

Lemma 3. [Property P.2] Consider a LFSR of memory r with feedback polynomial $c(D)$, $c_0 \neq 0 \neq c_r$, and $G(D) = \frac{g_0(D)}{c(D)}$. If $\gcd(g_0(D), c(D)) = 1$ then the period of $G(D)$ is the smallest integer p such that $c(D)$ divides $1 + D^p$.

Proof. Assume that $c(D)$ divides $1 + D^p$. Then

$$\frac{1+D^p}{c(D)} = a_0 + \cdots + a_{p-r}D^{p-r},$$

for some polynomial $a_0 + \ldots a_{p-r}D^{p-r}$. Therefore

$$G(D) = \frac{g_0(D)}{c(D)} = \frac{g_0(D)(a_0 + \cdots + a_{p-r}D^{p-r})}{1+D^p},$$

or

$$G(D)(1+D^p) = \underbrace{(a_0 + \cdots + a_{p-r}D^{p-r})g_0(D)}_{\text{polynomial of degree at most } (p-1).}.$$

This shows that $G(D)$ is periodic with period p.

Conversely assume that $G(D)$ has period p. Then

$$\frac{g_0}{c(D)} = G(D) = (a_0 + \cdots + a_{p-1}D^{p-1})(1 + D^p + D^{2p} + \ldots) = \frac{(a_0 + \cdots + a_{p-1}D^{p-1})}{1 + D^p},$$

from which we conclude that

$$g_0(D)(1 + D^p) = c(D)(a_0 + \cdots + a_{p-1}D^{p-1}).$$

From the condition $\gcd(g_0(D), c(D)) = 1$ and the unique factorization theorem we conclude that $c(D)$ divides $1 + D^p$. □

Definition 7. A maximum length LFSR (MLSR) of memory r is a LFSR of memory r whose least period p is equal to $2^r - 1$ for *all* nonzero initial states.

Lemma 4. [Property P.3] A necessary condition for a LFSR of memory r to have period $2^r - 1$ is that its feedback polynomial $c(D)$ be *irreducible*.

Proof. `Proof not complete!` Recall that by assumption $\deg(c) = r$, i.e., the degree of $c(D)$ is r. Assume to the contrary that $c(D)$ factors, lets say $c(D) = c_1(D)c_2(D)$ with $\deg(c_1) = r_1$, $\deg(c_2) = r_2$, $r = r_1 + r_2$. Let $g_0(D) = 1$. Then

$$G(D) = \frac{1}{c(D)} = \frac{1}{c_1(D)c_2(D)} = \frac{\alpha_1(D)}{c_1(D)} + \frac{\alpha_2(D)}{c_2(D)} = G_1(D) + G_2(D).$$

We see that in this case the sequence can be generated as the sum of two smaller LFSRs one of memory r_1 and the other of memory r_2. We claim that if G_1 has period p_1 and G_2 has period p_2 then G has a period $p = p_1 p_2$ (see Exercise 3.1). But $c_1(D)$ has memory r_1 and $c_2(D)$ has memory r_2 so that $p_1 \leq (2^{r_1} - 1)$ and $p_2 \leq (2^{r_2} - 1)$. Therefore $p = p_1 p_2 \leq (2^{r_1} - 1)(2^{r_2} - 1) < 2^r - 1$. □

Unfortunately irreducibility of $c(D)$ is only a *necessary* condition but it is not *sufficient*.

Example 9. The polynomial $c(D) = 1 + D + D^2 + D^3 + D^4$ is irreducible over the binary field, i.e., it can not be written as $c_1(D)c_2(D)$, for any pair $c_1(D)$, $c_2(D)$ of binary polynomials of degree at least one. However,

$$(1 + D + D^2 + D^3 + D^4)(1 + D) = 1 + D^5$$

which shows that $c(D)$ divides $1 + D^5$ and therefore has period 5! □

Although we will not prove this in class it is nevertheless comforting to know

Theorem 4. For every natural number r there exists a binary polynomial $c(D)$ of degree r such that $c(D)$ divides $1 + D^{2^r - 1}$ but such that $c(D)$ does not divide $1 + D^p$ for $p < 2^r - 1$. Such a polynomial is called *primitive* and when used as a feedback polynomial in a LFSR it generates a *maximum length LFSR*.

n	$c(D)$
1	$D+1$
2	D^2+D+1
3	D^3+D+1
4	D^4+D+1
5	D^5+D^2+1
6	D^6+D+1
7	D^7+D+1
8	$D^8+D^4+D^3+D^2+1$
9	D^9+D^4+1
10	$D^{10}+D^3+1$

Table 3.1: Table of some primitive polynomials of degree up to ten.

A small list of some primitive polynomials of degree up to ten is given in Table 3.1.

Lemma 5. [Property P.4] For a given primitive feedback polynomial $c(D)$ let $G_1(D) = \frac{g_1(D)}{c(D)}$ and $G_2(D) = \frac{g_2(D)}{c(D)}$. Then the two output sequences are simply delayed versions of each other. Further, the same is true for $G_1(D) + G_2(D)$. This is called the *delay and add property.*

Proof. See Exercise 3.2. □

RANDOMNESS PROPERTIES OF MLSR SEQUENCES

Lemma 6. [Balanced Property-R.1] Consider the output of a MLSR of memory r and period $p = 2^r - 1$. Then the relative frequency of zeros and ones is equal to $\frac{1}{2} - \frac{1}{2p}$ and $\frac{1}{2} + \frac{1}{2p}$, respectively, i.e., the relative frequencies are almost balanced.

Proof. Note that since the LFSR is a MLSR, i.e., its period is equal to $2^r - 1$, the state must take on all $2^r - 1$ non-zero binary r-tuples. Note that we can identify the output sequence with the sequence of contents of lets say the rightmost memory element. But this memory element will contain a zero $2^{r-1} - 1$ times and a one 2^{r-1} times. The result now follows if we divide by the period $2^r - 1$ to get the relative frequencies. □

Lemma 7. [Runlength Property-R.2] Consider the output of a MLSR of memory r and period $p = 2^r - 1$. Let $f(l)$ denote the relative frequency of runs (of zeros

or ones) of length l. Then

$$f(l) = \begin{cases} \frac{1}{2^l}, & l = 1,\ldots,r-2, \\ \frac{1}{2^{r-1}}, & l = r-1, r, \\ 0, & l > r. \end{cases}$$

Proof. The proof is not very long but slightly tricky. Consider a window of length r and consider sliding this window over the sequence. Each time we see in the rightmost $l+2$ positions of this window a pattern of the form $0\underbrace{*\cdots*}_{l\times}0$ or $1\underbrace{*\cdots*}_{l\times}1$ we add one to the counter which counts runs of length l. Since the sequence is periodic with period p we only have to shift our window over a length p and count the number of occurrences within this period. Note that this is equivalent to looking at all *states* of the MLSR. First let $l = 1,\ldots,r-2$. In this case out of all p state vectors exactly $22^{r-(l+2)}$ will trigger the event "a run of length l has occurred." Next look at runs of length $r-1$. In particular look at a run of $r-1$ zeros. Looking at the state vectors this happens if an only if the state $\underbrace{0\ldots0}_{(r-1)\times}1$ goes into the state $1\underbrace{0\ldots0}_{(r-1)\times}$. But this is the case since $\underbrace{0\ldots0}_{(r-1)\times}1$ can not go into the all zero state. Further, within the period p we see the state $\underbrace{0\ldots0}_{(r-1)\times}1$ exactly once. By a similar argument we see that there can not be a run of $(r-1)$ ones. Since $\underbrace{1\ldots1}_{(r-1)\times}0$ has to go to the all one state and can not go to $0\underbrace{1\ldots1}_{(r-1)\times}$. Finally look at runs of length r. Note first that there can not be runs longer than r since otherwise the LFSR would be stuck either in the all one or in the all zero state. And further within the period p we see the all one state exactly once but not the all zero state.

In order now to convert these occurrences into probabilities we just have to normalize by their total count. This total count is equal to

$$\sum_{l=1}^{r-2} 2\cdot 2^{r-(l+2)} + 2 = 2 + \sum_{i=1}^{r-2} 2^i = 2^{r-1}.$$

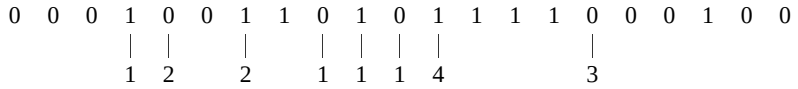
Dividing the individual occurrences by this sum results in the claim. □

Example 10. Choose $g_0(D) = 1$ and $c(D) = 1 + D^3 + D^4$. Using long division, we get that $G(D) = \frac{1}{1+D^3+D^4}$ has the following expansion

$$1 + D^3 + D^4 + D^6 + D^8 + D^9 + D^{10} + D^{11} + D^{15} + D^{18} + D^{19} + D^{21}\ldots$$

We know from Lemma 1 and 3 that this expansion is periodic with period at most $2^4 - 1 = 15$. From the explicit expansion of the first 22 terms we see that the

period is indeed 15. Running over one period we get the following picture.



Also shown are the beginnings of all runs as well as their lengths. We see that there are fours runs of length one, two runs of length two, one run of length three and one run of length 4, in exact agreement with Lemma 7. □

Lemma 8. [Correlation Property-R.3] Consider the output of a MLSR of memory r and period $p = 2^r - 1$. Compare now two shifted versions of this output. Then the two versions will agree at a given bit position with probability $\frac{2^{r-1}-1}{2^r-1}$.

Proof. See Exercise 3.3. □

6. SLIGHTLY MORE CAREFUL ANALYSIS

We are now ready to start a more careful analysis of a spread spectrum system.

Let the elements of the data sequence of a given user be denoted by u_i, with $u_i \in \{\pm 1\}$. Assume that we spread by a factor of N, i.e., one *symbol* period T is split into N *chip* periods T_c. Let x_n denote the elements of the data sequence up-sampled by a factor N, i.e.,

$$x_n := u_{\lfloor \frac{n}{N} \rfloor}, \;\; x_n \in \{\pm 1\}.$$

We multiply this upsampled data sequence x_n by a *complex valued* signature sequence

$$s_n = \frac{1}{\sqrt{2}}(s_n^I + j s_n^Q).$$

For the purpose of our analysis we will think of the components s_n^I and s_n^Q as independent flips of a fair coin, taking values in $\{\pm 1\}$. In practice, of course, these sequences are generated by LFSRs as discussed in the previous section. The sequence of elements $x_n s_n$ is sent through a pulse generator which generates pulses of energy $\sqrt{E_c}$ (where E_c is the energy per *chip* period) and the resulting pulse train is passed through a unit norm transmit filter $h(t)$. Therefore the baseband signal is given by

$$\sqrt{E_c} \sum_n x_n s_n h(t - nT_c).$$

Finally, this signal is up-converted. The whole transmitter is shown in Fig. 3.8. This is what is called a QPSK SS transmitter. We get the BPSK SS transmitter as a special case if we set $s_n^Q = 0$. We can represent this transmitter in a more compact way if we introduce the *equivalent impulse response*

$$g_i(t) = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} s_{iN+n} h(t - nT_c).$$

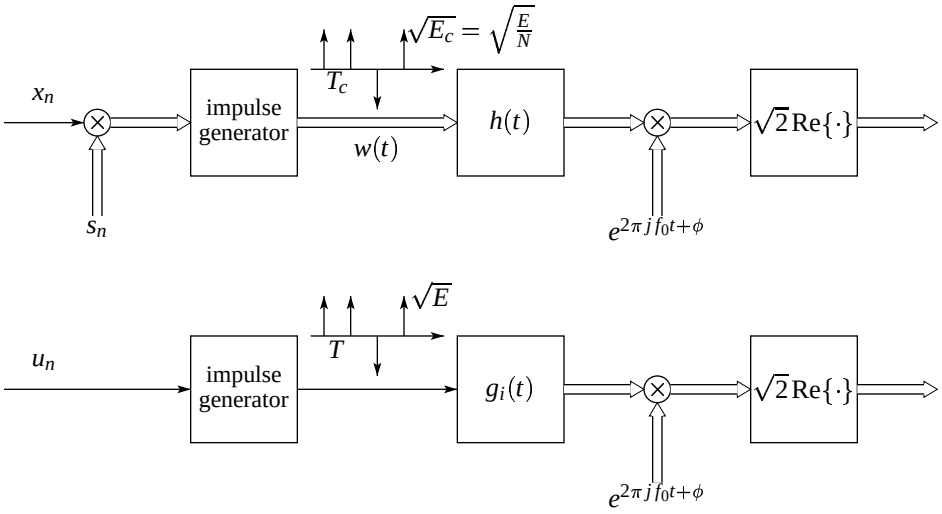


Figure 3.8: The basic QPSK SS transmitter. The function $g_i(t)$ is the equivalent impulse response.

It is a function of the time index i as well as the user index. In terms of this equivalent impulse response $g_i(t)$ the baseband signal can also be written as

$$\begin{aligned}\sqrt{E}\sum_i u_i g_i(t-iT) &= \sqrt{E}\sum_i u_i \frac{1}{\sqrt{N}}\sum_{n=0}^{N-1} s_{iN+n} h(t-iT-nT_c) \\ &= \sqrt{\frac{E}{N}}\sum_i \sum_{n=0}^{N-1} x_{iN+n} s_{iN+n} h(t-iT-nT_c) \\ &= \sqrt{\frac{E}{N}}\sum_n x_n s_n h(t-nT_c).\end{aligned}$$

The model depicted in Fig. 3.8 shows that QPSK SS is just antipodal modulation, where the impulse $g_i(t)$ changes for each data symbol u_i. Hence, the receiver for the AWGN channel is the matched filter receiver shown in Fig. 3.9.

This receiver minimizes the probability of error, assuming no intersymbol interference at the MF output (sample times). We will now investigate this receiver more closely. We assume that

$$\mathcal{R}_h(nT_c) = 0, \;\; n \text{ nonzero integer},$$

where

$$\mathcal{R}_h(\tau) = \int_{-\infty}^{\infty} h(t)h(t+\tau)dt.$$

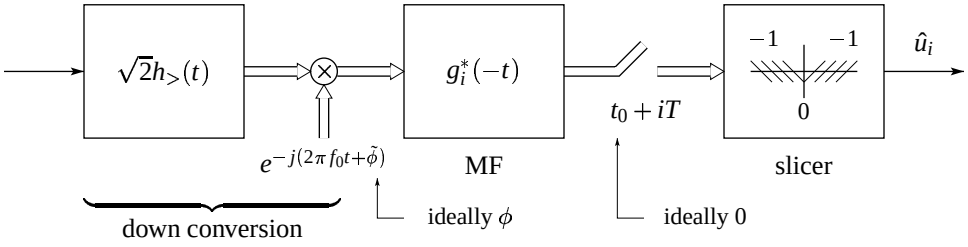


Figure 3.9: Receiver.

From now on we will consider the baseband equivalent model shown in Fig. 3.10.

6.1 Statistic of Matched Filter Output

In order to determine the performance of the system note that up to the input of the slicer the system is *linear*. Hence, the contributions stemming from (i) other users, (ii) the user of interest, (iii) background noise, can be considered separately. We will now consider each of these components in detail. In order to avoid a flood of indices we will use the following convention. There are k users in the system. Without loss of generality we will focus on the receiver for the first user. This user has an information sequence with elements u_i, an upsampled information sequence with elements x_n and a signature sequence with elements s_n. When we consider the output of the matched filter as a result of the transmission of some *other user* j, $j \in 2,\ldots,k$, we will denote the quantities pertaining to his transmission by $\tilde{}$. E.g., his upsampled data sequence will be denoted by $\tilde{x}_n$.

(i) Other users: Consider the situation depicted in Fig. 3.11. Let $\tilde{r}(t)$ denote the signal corresponding to some *other* user j, $1 < j \leq k$,

$$\tilde{r}(t) = \sqrt{\frac{E}{N}} \sum_n \tilde{x}_n \tilde{s}_n h(t - nT_c - \tilde{t}),$$

where $\tilde{t}$ accounts for the time difference between the users (we use the receiver time of the first user as reference). We have

$$\begin{aligned} b(t) &= \int_{-\infty}^{\infty} \tilde{r}(t-\tau)h(-\tau)d\tau \\ &= \int_{-\infty}^{\infty} \sqrt{\frac{E}{N}} \sum_m \tilde{x}_m \tilde{s}_m h(t-\tau-mT_c-\tilde{t})h(-\tau)d\tau \\ &= \sqrt{\frac{E}{N}} \sum_m \tilde{x}_m \tilde{s}_m \mathcal{R}_h(t-mT_c-\tilde{t}). \end{aligned}$$

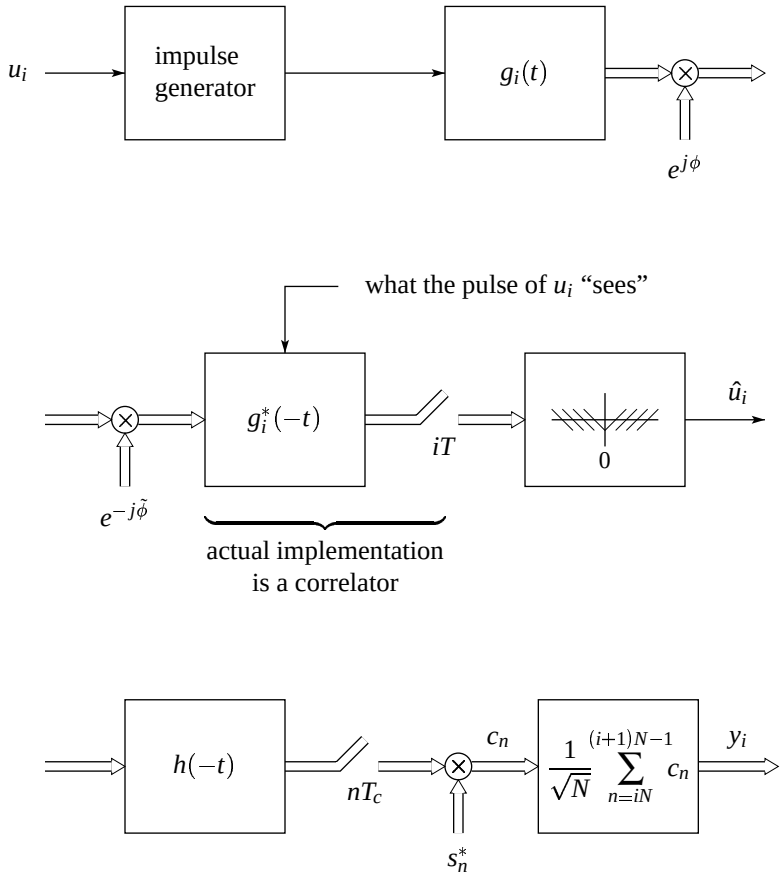


Figure 3.10: Equivalent baseband model.

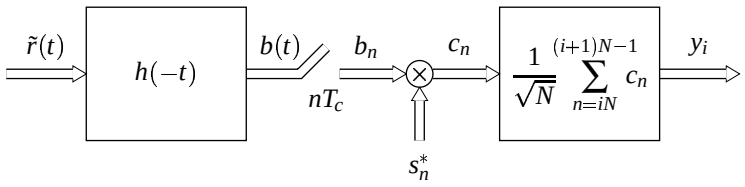


Figure 3.11: The output of the matched filter of the first user as a result of the received signal $\tilde{r}(t)$ from some *other* user j, $1 < j \leq k$.

Define $b_n := b(nT_c)$. Then

$$b_n = b(nT_c) = \sqrt{\frac{E}{N}} \sum_m \tilde{x}_m \tilde{s}_m \mathcal{R}_h((n-m)T_c - \tilde{t}).$$

The output of the matched filter is therefore

$$\begin{aligned}
y_i &= \frac{1}{\sqrt{N}} \sum_{n=iN}^{(i+1)N-1} c_n \\
&= \frac{1}{\sqrt{N}} \sum_{n=iN}^{(i+1)N-1} b_n s_n^* \\
&= \frac{1}{\sqrt{N}} \sum_{n=iN}^{(i+1)N-1} \sqrt{\frac{E}{N}} \sum_m \tilde{x}_m \tilde{s}_m \mathcal{R}_h((n-m)T_c - \tilde{t}) s_n^* \\
&\overset{l=n-m}{=} \sqrt{E} \sum_l \mathcal{R}_h(lT_c - \tilde{t}) \left[\frac{1}{N} \sum_{n=iN}^{(i+1)N-1} \tilde{x}_{n-l} \tilde{s}_{n-l} s_n^* \right] \\
&= \sqrt{E} \sum_l \mathcal{R}_h(lT_c - \tilde{t}) \xi_{i,l},
\end{aligned}$$

where

$$\xi_{i,l} := \frac{1}{N} \sum_{n=iN}^{(i+1)N-1} \underbrace{\tilde{x}_{n-l} \tilde{s}_{n-l} s_n^*}_{\beta_{n,l}}.$$

Note that $\mathbb{E}[\beta_{n,l}] = 0$ and that

$$\mathbb{E}\left[\beta_{n,l} \beta_{n,l}^*\right] = \mathbb{E}\left[\tilde{x}_{n-l} \tilde{x}_{n-l}^* \tilde{s}_{n-l} \tilde{s}_{n-l}^* s_n^* s_n\right] = 1.$$

Moreover, $\beta_{n,l}, \beta_{(n+1),l}, \ldots$ is an i.i.d sequence. Furthermore,

$$\mathbb{E}[\xi_{i,l} \xi_{i,k}^*] = \frac{1}{N^2} \sum_{n=iN}^{(i+1)N-1} \sum_{m=iN}^{(i+1)N-1} \underbrace{\mathbb{E}[\beta_{n,l} \beta_{m,k}^*]}_{1 \text{ iff } n=m \text{ and } l=k} = \frac{\delta_{l-k}}{N}.$$

This shows that $\xi_{i,l}$ and $\xi_{i,k}$ are uncorrelated if $l \neq k$. It follows that

$$\begin{aligned}\mathbb{E}[y_i y_i^*] &= E\sum_l \sum_k \mathcal{R}_h(lT_c - \tilde{t})\mathcal{R}_h(kT_c - \tilde{t}) \underbrace{\mathbb{E}[\xi_{i,l}\xi_{i,k}^*]}_{\frac{\delta_{l-k}}{N}} \\ &= \frac{E}{N}\sum_l [\mathcal{R}_h(lT_c - \tilde{t})]^2 .\end{aligned}$$

Since $\mathbb{E}[y_i] = 0$, the contribution of user j, $1 < j \leq k$, to the slicer input of the first receiver has zero mean and variance $\sigma_j^2 := \frac{E(j)}{N}\sum_l [\mathcal{R}_h(lT_c - t_j)]^2$.

We can assess the influence of the j-th user on the receiver of the first user in the following alternative way. For this assume that the relative time shift of the j-th user is uniformly distributed in $[0, T_c)$. Consider the complex pulse train

$$\tilde{w}(t) = \sqrt{\frac{E_c}{2}}\sum_n \tilde{x}_n(\tilde{s}_n^I + j\tilde{s}_n^Q)\delta(t - nT_c),$$

as shown in Fig. 3.8. As we have seen in Exercise 3.4, the *randomized* pulse train $\tilde{w}(t + \phi)$ is a WSS process with power spectral density equal to $\frac{E_c}{T_c}$. At the transmitter this process is now sent through the transmit filter which has impulse response $h(t)$, so that the output process has power spectral density equal to $\frac{E_c}{T_c}|H(f)|^2$. Consider now this WSS process at the input of the receiver. We first send this process through a filter with impulse response $h(-t)$. Therefore, at the point of the sampler we see a zero mean WSS process with power spectral density equal to $\frac{E_c}{T_c}|H(f)|^4$, call this process $b(t)$. If we sample $b(t)$ then the variance of the sample, call it b_n, is equal to $\mathcal{R}_b(0)$, which by definition, is equal to

$$\mathcal{R}_b(0) = \int_{-\infty}^{\infty} \mathcal{S}_b(f)df = \frac{E_c}{T_c}\int_{-\infty}^{\infty} |H(f)|^4 df.$$

Now consider y_i, which is equal to

$$\frac{1}{\sqrt{N}} \sum_{n=iN}^{(i+1)N-1} s_n^* b_n.$$

Since

$$\frac{1}{N}\mathbb{E}[b_n s_n^* b_m^* s_m] = \begin{cases} \frac{E_c}{NT_c}\int_{-\infty}^{\infty}|H(f)|^4 df, & n = m, \\ 0, & n \neq m, \end{cases}$$

it follows that y_i has variance equal to

$$\frac{E_c}{T_c}\int_{-\infty}^{\infty}|H(f)|^4 df = \frac{E}{NT_c}\int_{-\infty}^{\infty}|H(f)|^4 df \overset{\text{Parseval}}{=} \frac{E}{NT_c}\int_{-\infty}^{\infty}\mathcal{R}_h^2(\tau)d\tau.$$

This is compatible with our previous analysis, in which we got $\frac{E}{N}\sum_l |\mathcal{R}_h(lT_c - \tilde{t})|^2$. Indeed we just get the expected value over all possible shifts.

(ii) User of interest: The only difference to the preceding analysis is that now $\tilde{x}_n\tilde{s}_n = x_n s_n$ for all n. We claim that in this case

$$y_i = \sqrt{E}\mathcal{R}_h(t_1)u_i + z_i,$$

where z_i has zero mean and variance $\sigma_1^2 := \frac{E}{N}\sum_{l\neq 0}[\mathcal{R}_h(lT_c - t_1)]^2$. To see this claim first look at the case $l = 0$. Then we have

$$\begin{aligned}
\xi_{i,0} &= \frac{1}{N}\sum_{n=iN}^{(i+1)N-1} x_n s_n s_n^* \\
&= \frac{1}{N}\sum_{n=iN}^{(i+1)N-1} x_n \\
&= \frac{1}{N}\sum_{n=iN}^{(i+1)N-1} u_i \\
&= u_i.
\end{aligned}$$

For $l \neq 0$ the analysis proceeds exactly as before and the claim follows.

Example 11. Consider the case shown in Fig. 3.12. Assume that $0 \leq t_1 < T_c$.

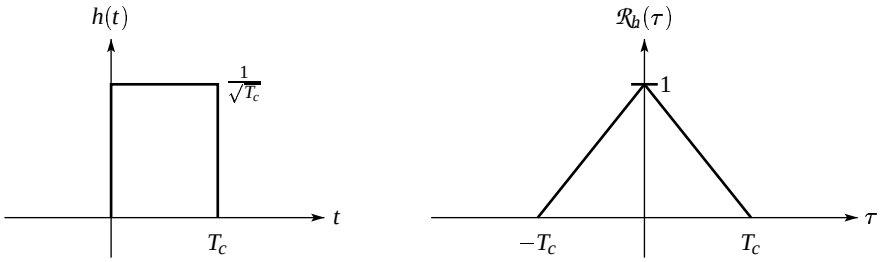


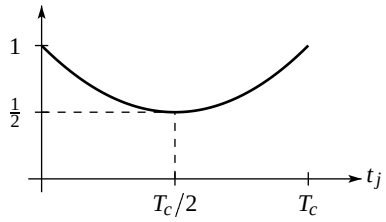
Figure 3.12: Rectangular transmit filter.

Then

$$\sqrt{E}\mathcal{R}_h(t_1) = \sqrt{E}(1 - \frac{t_1}{T_c}).$$

The variance of the noise stemming from some user j, $1 < j \leq k$, is then

$$\begin{aligned}
\frac{E}{N}\sum_l [\mathcal{R}_h(lT_c - t_j)]^2 &= \frac{E}{N}\left([\mathcal{R}_h(-t_j)]^2 + [\mathcal{R}_h(T_c - t_j)]^2\right) \\
&= \frac{E}{N}\left(\left(1 - \frac{t_j}{T_c}\right)^2 + \left(\frac{t_j}{T_c}\right)^2\right) \\
&= \frac{E}{N}\left(1 - \frac{2t_j}{T_c} + \frac{2t_j^2}{T_c^2}\right).
\end{aligned}$$



Alternative, we can calculate this quantity assuming that the relative time shifts are uniformly distributed in the range $[0, T_c)$, in which case we get

$$\frac{E}{NT_c}\int_{-\infty}^{\infty}|H(f)|^4 df \stackrel{\text{Parseval}}{=} \frac{E}{NT_c} 2\int_0^{T_c}(1-\frac{\tau}{T_c})^2 d\tau \stackrel{x=\tau/Tc}{=} \frac{2E}{N}\int_0^1 (1-x)^2 dx = \frac{2E}{3N}.$$

The variance stemming from the transmission of the user herself is equal to

$$\frac{E}{N}\sum_{l\neq 0}[\mathcal{R}_h(lT_c - t_1)]^2 = \frac{E}{N}[\mathcal{R}_h(T_c - t_1)]^2 = \frac{E}{N}\left(\frac{t_1}{T_c}\right)^2.$$

□

(iii) Finally, consider the effect of the background noise. Recall that the base-band equivalent noise has a (two-sided) power spectral density of N_0 and that the equivalent transmit filter $g_i(t)$ has unit norm. It follows that the contribution of the background noise to the slicer input of the first user is a complex valued Gaussian random variable with zero mean and variance σ_N^2 equal to

$$\sigma_N^2 = N_0\int_{-\infty}^{\infty}|g_i(t)|^2 dt = N_0.$$

We summarize: The output of the matched filter at time i has the form

$$y_i = \sqrt{E(1)}\mathcal{R}_h(t_1)u_i + z_i,$$

where z_i is a zero mean random variable with variance

$$\sigma^2 := \frac{E(1)}{N}\sum_{l\neq 0}|\mathcal{R}_h(lT_c - t_1)|^2 + \sum_{j=2}^{k}\frac{E(j)}{N}\sum_l |\mathcal{R}_h(lT_c - t_j)|^2 + N_0.$$

Alternatively, as discussed above, if we think of t_j as a random variable uniformly distributed in $[0, T_c)$, then we can replace $\frac{E(j)}{N}\sum_l |\mathcal{R}_h(lT_c - t_j)|^2$ by $\frac{E(j)}{NT_c}\int_{-\infty}^{\infty}|H(f)|^4 df$. In a typical scenario the interference caused by other users is the dominant effect. In this case we will assume that the variance of the noise is given by

$$\sigma^2 = \left(\sum_{j=2}^{k}\frac{E(j)}{NT_c}\right)\int_{-\infty}^{\infty}|H(f)|^4 df.$$

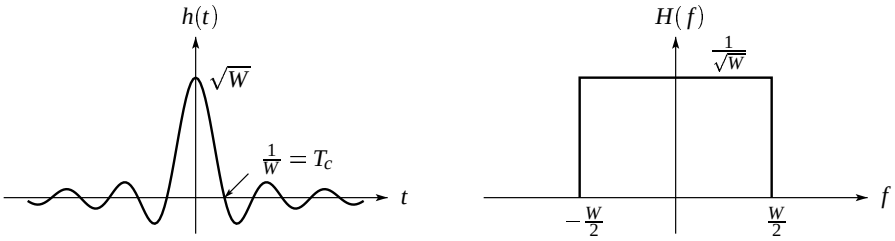


Figure 3.13: Specific transmit filter.

Example 12. Let $h(t)$ be as given in Fig. 3.13. If $T_c = \frac{1}{W}$ then there is no ISI. Then we have

$$\int_{-\infty}^{\infty} |H(f)|^4 df = W\frac{1}{W^2} = \frac{1}{W} = T_c.$$

□

Example 13. [Optimality of Transmit Filter With Flat Spectrum] Suppose $|H(f)| = 0$ for $|f| > \frac{W}{2}$ and $\int_{-\infty}^{\infty} h^2(t)dt = 1$. Then by the Schwarz inequality[3] we get

$$\left(\int_{-\frac{W}{2}}^{\frac{W}{2}} |H(f)|^4 df\right)\left(\int_{-\frac{W}{2}}^{\frac{W}{2}} df\right) \geq \left[\int_{-\frac{W}{2}}^{\frac{W}{2}} |H(f)|^2 df\right]^2 = 1.$$

We conclude that

$$\int_{-\frac{W}{2}}^{\frac{W}{2}} |H(f)|^4 df \geq \frac{1}{W} = T_c,$$

with equality iff $|H(f)|^2 = \frac{1}{W}$ for $|f| \leq \frac{W}{2}$. Hence, for bandlimited spectrum the constant $H(f)$ minimizes the variance stemming from signals from other users.

□

INTERPRETATION OF VARIOUS INTERFERENCE TERMS

Recall that the received signal of the user of interest is equal to

$$r(t) = \sqrt{E}\sum_i u_i g_i(t - iT).$$

[3]Let f and g be real valued square integrable functions and define $\|f\| := \sqrt{\int f^2}$. For any real valued pair of numbers α and β define $h = \alpha f - \beta g$. We then have

$$0 \leq \|h\|^2 = \alpha^2\|f\|^2 + \beta^2\|g\|^2 - 2\alpha\beta\int fg.$$

Let $\alpha := \|g\|$ and $\beta := \|f\|$. Then this reads

$$0 \leq \|h\|^2 = 2\|f\|^2\|g\|^2 - 2\|f\|\|g\|\int fg.$$

If $\|f\| \neq 0 \neq \|g\|$ we conclude that $\int fg \leq \|f\|\|g\|$, and it is easy to check that the same conclusion holds if either $\|f\| = 0$ or $\|g\| = 0$.

Let's assume that $N = 4$ and that we use a rectangular transmit filter. A short segment of the transmitted signal is then shown in Fig. 3.14. Let's first assume

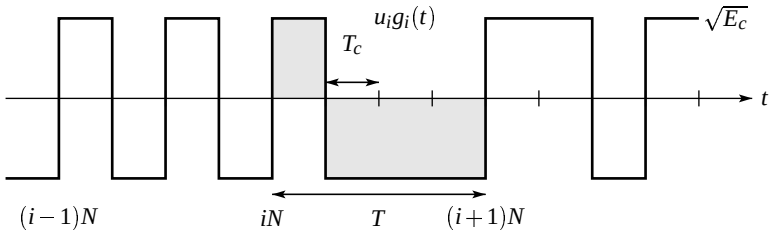
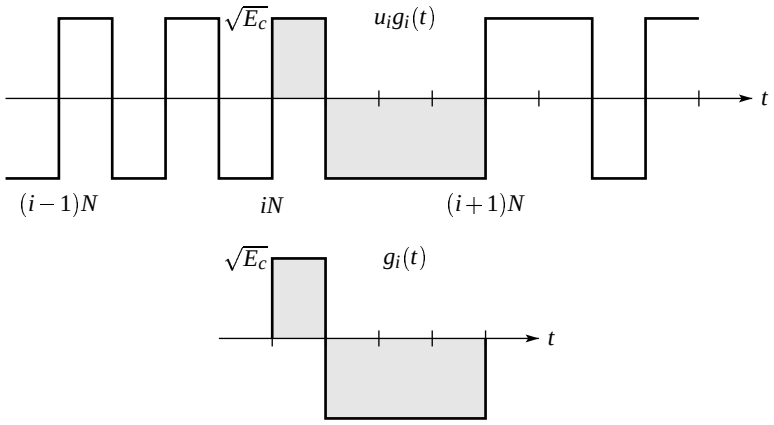
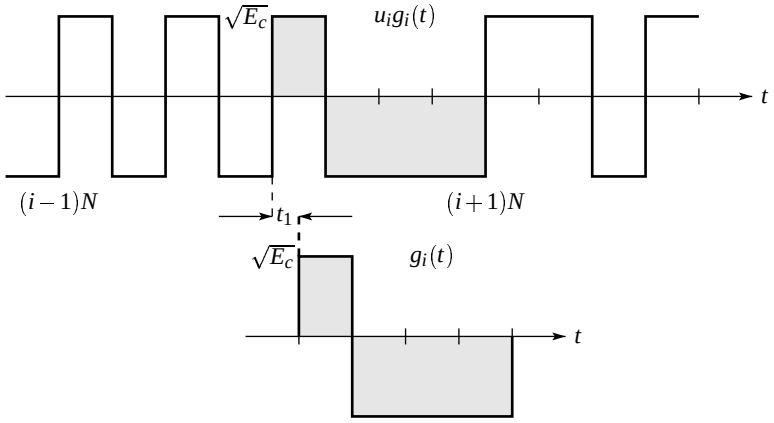


Figure 3.14: A short segment of the transmitted signal $r(t) = \sqrt{E}\sum_i u_i g_i(t - iT)$.

that $t_1 = 0$, i.e., that we have perfect synchronization. For the l-th time step the receiver then correlates the received signal $\sqrt{E}\sum_i u_i g_i(t - iT)$ with $g_l(t - lT)$. (in the figure we assume that $u_l = 1$.



Since we assumed that $\mathcal{R}_b(nT_c) = 0$, for n a non-zero integer, it follows that the result of this correlation is equal to $\sqrt{E}u_i$, which is just the desired signal. Assume next that we have an offset of t_1.



In this case each component of $g_l(t-lT)$, i.e., each signal $\frac{1}{N}h(t-lT-nTc)$ will have a correlation with its counterpart in the signal $r(t)$ of $\sqrt{E}N\mathcal{R}_h(t_1)u_i$. Since there N such contributions, the received signal in this case is equal to $\sqrt{E}\mathcal{R}_h(t_1)u_i$. The factor $\mathcal{R}_h(t_1)$ is always less than one and represents the penalty for non-perfect synchronization. In addition, each component of $g_l(t-lT)$ will, in general, have a non-zero correlation also with any of the other components of $r(t)$. This gives rise to the term $\frac{E}{N}\sum_{l\neq 0}|\mathcal{R}_h(lT_c-t_1)|^2$.

A very similar picture applies if we want to interpret the variance terms which stem from the interference from other users.

6.2 Probability of Error Analysis

Let's perform a sanity check. Let's assume that the total variance of the noise is given by the approximation

$$\sigma^2 = \left(\sum_{j=1}^{k}\frac{E(j)}{NT_c}\right)\int_{-\infty}^{\infty}|H(f)|^4 df,$$

i.e., we assume that the background noise as well as the noise stemming from the transmission of the user himself is negligible. If we assume that we transmit over a bandpass channel of bandwidth W then the optimal transmit filter is the one which is flat over the whole frequency region as we saw in Example 13. Further, we assume that the received energy per symbol of all users are identical and equal to E. In this case the variance of the noise is easily seen to be $(k-1)\frac{E}{N}$. The signal of interested is $\pm\sqrt{E}$ (here we assume that we have perfect synchronization so that $t_1 = 0$). If we further assume that the noise is complex symmetric Gaussian then the variance *per dimension* is equal to $(k-1)\frac{E}{2N}$ and if we were to decide on a

symbol by symbol basis we would incur a bit error probability of

$$Q\left(\frac{d}{2\sigma}\right) = Q\left(\sqrt{\frac{2N}{(k-1)}}\right).$$

Consider an antipodal transmission scheme with energy E and background noise of power spectral density equal to $\frac{N_0}{2}$. In this case the error probability is equal to

$$Q\left(\sqrt{\frac{2E}{N_0}}\right).$$

Therefore, $\frac{N}{k-1}$ corresponds to $\frac{E}{N_0}$. Assume now that we employ a code with a *redundancy* of r. E.g., if $r = \frac{1}{2}$ this means that for every *information bit* there are actually two *transmitted* bits. The energy expended per *information* bit is then equal to $E_b = \frac{1}{r}E$, so that $\frac{E_b}{N_0} = \frac{E}{rN_0} = \frac{N}{r(k-1)}$. Assume that our *information rate* is R bits per seconds and that $T_c = \frac{1}{W}$. The *transmission rate* is then equal to $\frac{R}{r}$ bits per second and $T = \frac{r}{R}$. It follows that $N = \frac{T}{T_c} = \frac{rW}{R}$ and therefore

$$\frac{E_b}{N_0} = \frac{E}{rN_0} = \frac{N}{r(k-1)} = \frac{W/R}{k-1}.$$

As discussed in the introduction of this section, depending on e.g. the channel model, the coding scheme, or the desired probability of error there exists a number $\left(\frac{E_b}{N_0}\right)_{\text{req.}}$ so that our condition on the number of supportable users will be

$$\frac{W/R}{k-1} \geq \left(\frac{E_b}{N_0}\right)_{\text{req.}} \Leftrightarrow (k-1) \leq \frac{W/R}{\left(\frac{E_b}{N_0}\right)_{\text{req.}}}.$$

This is the same result we got with our rough estimate in the introduction.

EXERCISES

3.1. Let $G_1(D)$ and $G_2(D)$ be generating functions of two sequences and define $G(D) = G_1(D) + G_2(D)$. Show that if G_1 has period p_1 and G_2 has period p_2 then G has period $p = p_1 p_2$.

3.2. Proof Lemma 5.

3.3. Proof Lemma 8.

3.4. Consider all the following steps carefully. The sequence $w(t)$ shown in Fig. 3.8 is a complex-valued random pulse train. Ideally, each pulse has zero

width and unit "area". In practice one has to be content with *finite* width pulses. To see the effect of such a finite width pulse assume that

$$w(t) = \sqrt{\frac{E_c}{2}} \sum_n x_n (s_n^I + j s_n^Q) f_\Delta(t - nT_c),$$

where

$$f_\Delta(t) := \begin{cases} \frac{1}{\Delta}, & |t| \le \frac{\Delta}{2}, \\ 0, & |t| > \frac{\Delta}{2}. \end{cases}$$

The autocorrelation of $w(t)$ is then

$$\begin{aligned} \mathcal{R}_w(t,s) &= \mathbb{E}[w(t)w^*(s)] \\ &= \mathbb{E}\left[w^I(t)w^I(s) + w^Q(t)w^Q(s)\right] + j\mathbb{E}\left[-w^I(t)w^Q(s) + w^Q(t)w^I(s)\right] \\ &= \mathbb{E}\left[w^I(t)w^I(s) + w^Q(t)w^Q(s)\right]. \end{aligned}$$

The second step follows since

$$\begin{aligned} \mathbb{E}\left[w^I(t)w^Q(s)\right] &= \frac{E_c}{2}\mathbb{E}\left[\left(\sum_n x_n s_n^I f_\Delta(t - nT_c)\right)\left(\sum_m x_m s_m^Q f_\Delta(s - mT_c)\right)\right] \\ &= \frac{E_c}{2}\sum_{n,m} f_\Delta(t - nT_c) f_\Delta(s - mT_c)\mathbb{E}\left[x_n x_m \underbrace{\mathbb{E}\left[s_n^I s_m^Q\right]}_{0}\right] \\ &= 0, \end{aligned}$$

and in the same way we see that $\mathbb{E}\left[w^Q(t)w^I(s)\right] = 0$. Since

$$\mathbb{E}\left[s_n^I s_m^I\right] = \mathbb{E}\left[s_n^Q s_m^Q\right] = \delta_{n,m},$$

we get $\mathcal{R}_w(t,s) = E_c \sum_n f_\Delta(t - nT_c) f_\Delta(s - nT_c)$. Note that $w(t)$ is not WSS but cyclostationary with period T_c. But if we add a random phase, i.e., if we consider the random process $z(t) := w(t + \phi)$ with ϕ uniformly distributed over $[0, T_c)$ then for this process we get

$$\begin{aligned} \mathcal{R}_z(t,s) &= \mathbb{E}\left[w^I(t)w^I(s) + w^Q(t)w^Q(s)\right] \\ &= 2\mathbb{E}\left[w^I(t)w^I(s)\right] \\ &= E_c\mathbb{E}\left[\sum_n f_\Delta(t + \phi - nT_c) f_\Delta(s + \phi - nT_c)\right] \\ &= \frac{E_c}{T_c}\int_{-\infty}^{\infty} f_\Delta(r + t) f_\Delta(r + s) dr \\ &= \frac{E_c}{T_c}\int_{-\infty}^{\infty} f_\Delta(r) f_\Delta(r + (s - t)) dr \\ &= \mathcal{R}_z(t - s), \end{aligned}$$

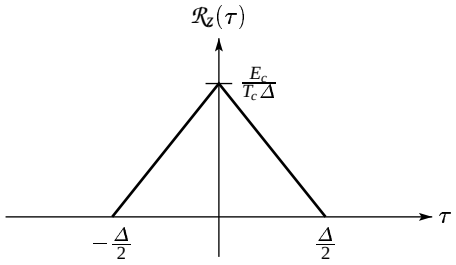


Figure 3.15: Autocorrelation function of the random pulse train $z(t) := w(t+\phi)$ generated by finite width pulses. Hereby, ϕ is a random offset which is uniformly distributed over $[0, T_c)$.

which shows that $w(t+\phi)$ is a WSS random process. Further, the autocorrelation function $\mathcal{R}_z(t-s)$ is easily determined to be the one shown in Fig. 3.15. As the pulse width Δ converges to zero this autocorrelation function $\mathcal{R}_z(\tau)$ converges to $\frac{E_c}{T_c}\delta(\tau)$.

3.5. This exercise deals with a simple yet powerful bounding technique called the Chernoff bound. Let $X_1, \ldots, X_n$ be a set of n i.i.d. random variables. We are interested to bound $\Pr\{\sum_{i=1}^n X_i > \alpha\}$. Verify the following steps, where $s > 0$,

$$\begin{aligned}
\Pr\{\sum_{i=1}^n X_i > \alpha\} &= \Pr\{e^{s\sum_{i=1}^n X_i} > e^{s\alpha}\} \\
&\leq \min_{s>0} \frac{\mathbb{E}[e^{s\sum_{i=1}^n X_i}]}{e^{s\alpha}} \\
&= \min_{s>0} \mathbb{E}[e^{sX_1}]^n e^{-s\alpha}.
\end{aligned}$$

Now let X be binary taking values in $\{\pm 1\}$ with equal probability. How do you have to choose α (as a function of n) such that you get an exponential bound, i.e., a bound of the form e^{-nc} for some strictly positive constant c. How do you have to choose α (as a function of n) such that the right hand side is a constant but can be made arbitrarily small? Next let $X_i \sim \mathcal{N}(0, \sigma^2)$. What do you get now? For the Gaussian case can you determine the probability exactly? How do the results compare?

3.6. Assume that we allow each user to scale his input signal by a factor α, $\alpha > 1$. In the interference limited case, how does this effect the system performance?

3.7. In the main text we assumed that the signature sequences s_n are complex valued of the form $s_n = \frac{1}{\sqrt{2}}(s_n^Q + j s_n^I)$, where s_n^Q, s_n^I are sequences taking values in $\{\pm 1\}$. Assume now that s_n is *real valued* taking values in $\{\pm 1\}$. Retrace the steps of our analysis. How large is the noise variance? Assume that the system is

interference limited. How does the number of supportable users change compared to the complex valued case?

3.8. In our analysis we allowed complex valued signatures but we restricted our data sequence to take elements in $\{\pm 1\}$. Assume now that we allow our data sequence to take on the four values $\frac{1}{\sqrt{2}}\{\pm 1 \pm j\}$. Assume again the interference limited case. Assume that no coding is used and that we decide symbol by symbol. Can we support a higher rate with such a system compared to the system we analysed in class?

3.9. In class we investigated the use of spread-spectrum as a multiple-access problem. In this problem we will see that spread-spectrum can also be used to make signals less sensitive against jammers (malicious or spurious interfering signals.)

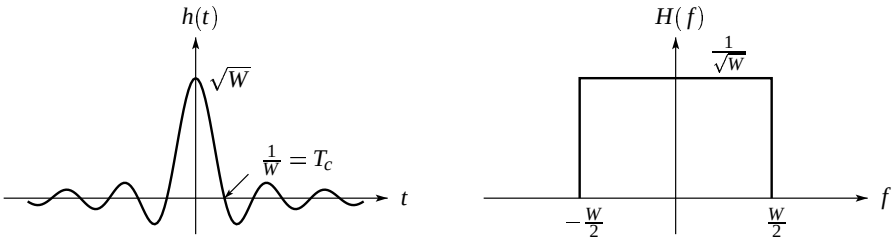
Assume we have a spread-spectrum system with a *single user*. This user transmits the signal

$$x(t) = \sqrt{\frac{E}{N}} \sum_n x_n s_n h(t - nT_c) = \sqrt{E} \sum_i u_i g_i(t - iT),$$

where

$$g_i(t) = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} s_{iN+n} h(t - nT_c),$$

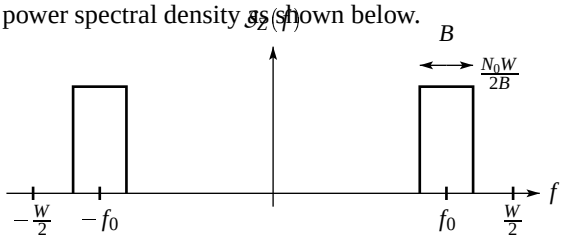
and where s_n is an i.i.d. sequence of complex valued random variables with zero mean and unit modulus (norm). Assume further that $h(t)$ is as shown below.



Assume that the received signal is equal to

$$y(t) = \sqrt{\frac{E}{N}} \sum_n x_n s_n h(t - nT_c) + Z(t) = \sqrt{E} \sum_i u_i g_i(t - iT) + Z(t)$$

where $Z(t)$ is a WSS complex-valued circularly-symmetric Gaussian process with zero mean and power spectral density as shown below.



Let y_i be the sampled output at the matched filter, i.e.,

$$y_i := \int y(t) g_i^*(t - iT)dt = \int x(t) g_i^*(t - iT)dt + \int Z(t) g_i^*(t - iT)dt.$$

1. Show that the total power of the noise $Z(t)$ is equal to WN_0.

2. We claim that
$$y_i = \sqrt{E}u_i + z_i,$$
where z_i is a a sequence of i.i.d. random variables which are complex-valued, circularly-symmetric Gaussian with zero mean and variance N_0. To prove the claim proceed as follows.

   (a) Show that the signal portion $\int x(t) g_i^*(t - iT)dt$ is equal to $\sqrt{E}u_i$.

   (b) Show that $z_i = \int Z(t) g_i^*(t - iT)dt$ is equal to
$$z_i := \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} s_{iN+n}^* W_{iN+n}, \tag{3.3}$$
   where
$$W_{iN+n} := \int Z(t) h(t - iT - nT_c)dt. \tag{3.4}$$

   (c) Starting from (3.4), show that the random variables W_{iN+n} are (dependent) complex-valued circularly-symmetric Gaussian random variables with zero mean and variance N_0.

   (d) Conclude from (3.3) that z_i is a sequence of i.i.d. random variables which are complex-valued, circularly-symmetric Gaussian with zero mean and variance N_0.

   This shows that even though the interfering signal is concentrated around a particular frequency f_0, the effect is equivalent to the effect of an interferer which spreads its power uniformly over the whole frequency band of interest.

4

HOW TO GET CLOSE TO CAPACITY: CLUES FROM INFORMATION THEORY

In the information theory class you have learned all about channel capacity for various channels (like the binary symmetric channel or the discrete-time additive white Gaussian noise channel) whereas in this digital communications course we have been mostly concerned with particular transmissions schemes for the linear time-invariant Gaussian noise channel, in particular the class of pulse-amplitude modulation schemes. In this chapter we will explore somewhat the connections between these two views. In particular, we will see that information theory not only gives us bounds on the rates at which we can transmit reliably but also indicates how these rates can be approached in practice with low-complexity schemes.

For the most part we will only discuss linear time-invariant channels with additive Gaussian noise but similar statements can be made for more general channels like e.g. *fading* channels.

1. THE LINEAR TIME-INVARIANT GAUSSIAN CHANNEL

Consider the linear time-invariant channel with additive Gaussian noise described by

$$Y(t) = X(t) * h(t) + Z(t).$$

Hereby, $X(t)$ is the input to the channel and we restrict this input via either a peak or an average power constraint P. The channel is characterized by its impulse response $h(t)$ or, equivalently, its spectrum $H(f)$, and $Z(t)$ is a real-valued wide sense stationary Gaussian process with double-sided power spectral density equal to $\frac{N_0}{2}$. Note that we assumed that the power spectral density is flat. This entails no essential loss of generality as can be seen by the following argument. Consider

a general power spectral density $\mathcal{S}_Z(f)$. First, note that we can freely change $\mathcal{S}_Z(f)$ outside the support of $H(f)$, i.e., for those frequencies for which $H(f) = 0$. Further, within the support of $H(f)$ we can assume that $\mathcal{S}_Z(f)$ is strictly positive almost everywhere since otherwise if $\mathcal{S}_Z(f)$ were zero over a measurable range within the support of $H(f)$ then it is easy to see that the capacity of such a channel would be infinite. In practice we are always transmitting over a finite frequency range, and therefore with the extra assumption that $\mathcal{S}_Z(f)$ is strictly bounded away from zero in this finite range of interest it follows that there exists a whitening filter. Since this whitening filter is further invertible it follows that, instead of the original channel we can consider the concatenation of the original channel with the whitening filter and that this new channel has the same capacity (since the operation of whitening the noise is invertible).

We are concerned with the following two basic questions: (i) What is the maximal rate of information at which we can transmit reliably over this channel. (ii) How can we approach this rate by low complexity techniques.

Our derivation will be quite heuristic but it essentially tells the right story. A rigorous derivation can be found in the book by Gallager.

2. CAPACITY OF THE LINEAR TIME-INVARIANT GAUSSIAN CHANNEL

We will start by answering question (i) concerning the capacity. We will do so by relating the capacity of the general linear time-invariant Gaussian noise channel to the capacity of the much simpler discrete-time additive white Gaussian noise channel.

2.1 DISCRETE TIME GAUSSIAN CHANNEL

First recall from your information theory class that if $Z \sim \mathcal{N}(0, \sigma^2)$ then its differential entropy, call it $h(Z)$, is equal to

$$\begin{aligned}
h(Z) &= \int -p_Z(x) \ln p_Z(x) dx \\
&= \int p_Z(x) \left[\frac{1}{2\sigma^2} x^2 - \ln \frac{1}{\sqrt{2\pi\sigma^2}} \right] dx \\
&= \frac{1}{2\sigma^2} \int p_Z(x) x^2 dx + \ln \sqrt{2\pi\sigma^2} \int p_Z(x) dx \\
&= \frac{1}{2} + \frac{1}{2} \ln(2\pi\sigma^2) \\
&= \frac{1}{2} \ln(2\pi e\sigma^2) \text{ nats} \\
&= \frac{1}{2} \log(2\pi e\sigma^2) \text{ bits.}
\end{aligned}$$

Further, if X is any zero-mean continuous random variable with density $p_X(x)$ and variance σ^2 then $h(X) \leq h(Z) = \frac{1}{2}\log_2(2\pi e\sigma^2)$ bits since

$$\begin{aligned} 0 &\leq D(p_X|p_Z) \\ &= \int p_X(x)\ln p_X(x)dx - \int p_X(x)\ln p_Z(x)dx \\ &= -h(X) + \int p_X(x)\left[\frac{1}{2\sigma^2}x^2 - \ln\frac{1}{\sqrt{2\pi\sigma^2}}\right]dx \\ &= -h(X) + \int p_Z(x)\left[\frac{1}{2\sigma^2}x^2 - \ln\frac{1}{\sqrt{2\pi\sigma^2}}\right]dx \\ &= -h(X) + h(Z). \end{aligned}$$

Consider now the discrete-time channel

$$Y_n = X_n + Z_n,$$

where X_n is the real-valued channel input with average energy constraint $\mathbb{E}[X_n^2] \leq E$ and Z_n is an i.i.d. sequence of real-valued zero-mean Gaussian random variables with variance σ^2.

In the information theory class you have learned that the capacity of this channel is equal to

$$\begin{aligned} C &= \max_{p(x):\mathbb{E}[X^2]\leq E} I(X;Y) \\ &= \max_{p(x):\mathbb{E}[X^2]\leq E} \{h(Y) - h(Y|X)\} \\ &= \max_{p(x):\mathbb{E}[X^2]\leq E} \{h(Y) - h(X+Z|X)\} \\ &= \max_{p(x):\mathbb{E}[X^2]\leq E} \{h(Y) - h(Z|X)\} \\ &= \max_{p(x):\mathbb{E}[X^2]\leq E} \{h(X+Z)\} - h(Z) \\ &= \frac{1}{2}\log\left(2\pi e(E+\sigma^2)\right) - \frac{1}{2}\log\left(2\pi e\sigma^2\right) \\ &= \frac{1}{2}\log\left(1+\frac{E}{\sigma^2}\right) \text{ bits per channel use.} \end{aligned}$$

Similarly, if we assume that the X_n are complex-valued with average energy constraint $\mathbb{E}[X_n^2] \leq 2E$ and that Z_n is an i.i.d. sequence of complex-valued circularly-symmetric zero-mean Gaussian random variables with variance $2\sigma^2$ then the capacity is $\log\left(1+\frac{E}{\sigma^2}\right)$ bits per channel use. To see this we can either retrace our previous steps and calculate the mutual information directly or we can recall that in the circularly symmetric case we can think of one complex dimension as simply two real-valued dimensions.

2.2 THE STANDARD BASEBAND CHANNEL

Consider now the standard baseband channel

$$Y(t) = (X(t) + Z(t)) * h(t),$$

where $h(t)$ is the impulse response of a baseband channel with bandwidth W.

Assume we transmit over this channel over a time period of T seconds. It is well-known that there are roughly $2WT$ dimensions in this space and, therefore, there are roughly $2W$ dimensions per second. It follows that the average energy per dimension we can expand is equal to $\frac{P}{2W}$. Hence, the capacity of this channel is equal to

$$2W\frac{1}{2}\ln\left(1+\frac{\frac{P}{2W}}{\frac{N_0}{2}}\right) = W\ln\left(1+\frac{P}{N_0W}\right) \text{ bits per second.}$$

This is Shannon's famous formula. If we let the bandwidth go to infinity and use the fact that $ln(1+x) = x + O(x^2)$ we see that

$$\lim_{W\to\infty} W\ln\left(1+\frac{P}{N_0W}\right) = \frac{P}{N_0}.$$

Next, assume that the bandwidth is $\frac{W}{2}$ and that the noise is complex with two sided power spectral density equal to N_0 and that the input is complex-valued with power equal to $2P$. In this case we can relate this to the complex valued channel and we get the same result (half the number of channel instances but twice the capacity per channel use).

3. THE UNCONSTRAINED CAPACITY VERSUS THE CAPACITY OF SPECIFIC SIGNALING SETS

Let's consider again the simplest continuous channel, namely the standard baseband channel. In this case we have seen that by appropriate signaling and sampling we can bring this channel model back to a discrete-time channel model. There still remains one problem however. Looking back at the derivation of the capacity formula it seems that we have to use a signaling alphabet which is "Gaussian." This is quite in contrast to all the examples we have seen in class so far, where we have used simple input alphabets like antipodal signalling or some simple PSK or QAM schemes. From a practical point of view it is clearly desirable to use such a simple signaling set. How much do we loose if we do that?

3.1 THE CAPACITY OF SPECIFIC SIGNALING SETS

Let's start with the simplest one-dimensional signaling scheme – antipodal signaling, i.e., we assume that $X_n \in \{\pm\sqrt{E}\}$ and that Z_n is real-valued Gaussian with

variance σ^2. Further, we assume that the points $\pm\sqrt{E}$ have equal prior. The capacity of such a signaling scheme is then easily calculated to be

$$\begin{aligned} C_{2-\text{PAM}}\left(\frac{E}{\sigma^2}\right) &= I(X;Y) \\ &= h(Y) - h(Y|X) \\ &= h(Y) - h(Z) \\ &= -\int p_Y(y)\log p_Y(y)dy - \frac{1}{2}\log(2\pi e\sigma^2), \end{aligned}$$

where

$$p_Y(y) = \sum_x p_{Y,X}(y,x) = \frac{1}{\sqrt{8\pi\sigma^2}}\left(e^{-\frac{(y-\sqrt{E})^2}{2\sigma^2}} + e^{-\frac{(y+\sqrt{E})^2}{2\sigma^2}}\right).$$

Unfortunately, there is no elementary solution to the above integral but the capacity $C_{2-\text{PAM}}\left(\frac{E}{\sigma^2}\right)$ can be evaluated to any desired degree of accuracy numerically. The result is shown in Fig. 4.1 We see from the above figure that as long as the

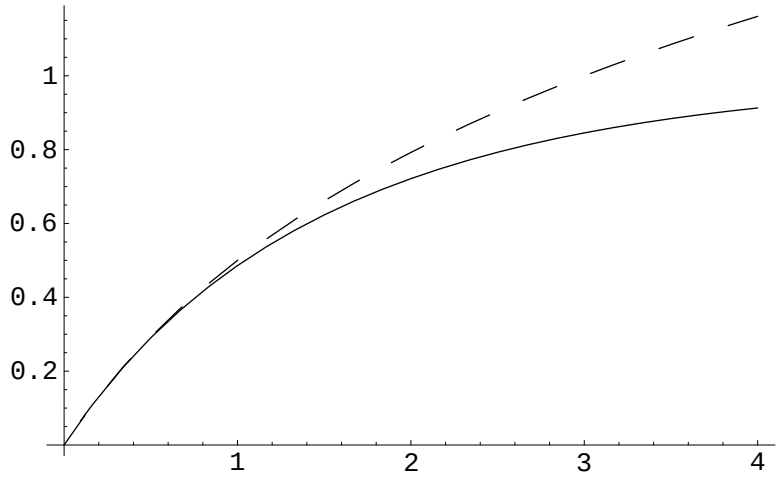


Figure 4.1: Maximal achievable rate of reliable transmission for antipodal signaling (solid line) and the capacity of the AWGN channel (dashed line) as a function of $\frac{E}{\sigma^2}$.

signal-to-noise ratio $\frac{E}{\sigma^2}$ is not too large (≤ 1) the loss which we incur by restricting ourself to antipodal signaling is very small. For large signal-to-noise ratios this loss can be become arbitrarily large. In this case we have to use larger signal constellations.

Therefore, lets next look at two-dimensional constellations, i.e., we can think of X_n as complex valued with average energy equal to $2E$ and Z_n is now complex-valued and circularly-symmetric with variance equal to $2\sigma^2$. Let the signal set

be denoted by $\mathcal{S} = \{s_1, \ldots, s_{2^k}\}$, $s_i \in \mathbb{C}$, where for convenience we have assumed that the number of signal points is a power of two. Standard choices are PSK constellations where the points are arranged on a circle or QAM where the points are arranged on a square grid. As before the capacity for a given constellation $\mathcal{S}$ can be written down in terms of

$$\begin{aligned} C_{\mathcal{S}} &= I(X;Y) \\ &= h(Y) - h(Y|X) \\ &= h(Y) - h(Z) \\ &= -\int p_Y(y) \log p_Y(y) dy - \log(2\pi e\sigma^2), \end{aligned}$$

where

$$p_Y(y) = \frac{1}{2\pi\sigma^2 2^k} \sum_{i=1}^{2^k} e^{-\frac{(y^R - s_i^R)^2}{2\sigma^2}} e^{-\frac{(y^I - s_i^I)^2}{2\sigma^2}}.$$

ToDo: Insert plots for various examples.

3.2 Multilevel Modulation and the Chain Rule of Mutual Information

Assume now that we use a larger signal constellation. For simplicity we will assume that our signal constellation is two-dimensional and we will use as our running example the 16-QAM constellation shown in Fig. 1.3 but it will be clear that the same ideas easily apply to a large class of signal constellations.

Assume that our signal constellation $\mathcal{S}$ contains 2^k points. It is then natural to label these points by k bits, call them $(X^1, \cdots, X^k)$. One such particular labeling is shown in Fig. 1.3 for the case $k = 4$. Let S_n, $S_n \in \mathcal{S}$, denote the symbol which is transmitted at time n. We formally specify this labeling by introducing a map ψ, $\psi : \hat{\gamma}^k \to \mathcal{S}$ which maps a k-tuple of bits $(X^1, \cdots, X^k)$ into a point of the constellation $\mathcal{S}$. In general there are $2^k!$ possible such labelings and we will see shortly how our choice of labeling affects the overall system performance. Our channel model is now

$$Y_n = S_n + Z_n,$$

where the noise is Gaussian with independent components each with variance σ^2. Hereby we assume the bits to be equally probable. To determine the capacity note that

$$\begin{aligned} I(S;Y) &= I(X^1, \ldots, X^k; Y) \\ &= \sum_{i=1}^{k} I(X^i; Y | X_1, \ldots, X^{i-1}), \end{aligned}$$

where the last step is simply the well-known *chain-rule* of mutual information. It is crucial to notice that the mutual information is *independent* of the labeling map

ψ but that the partitiong of the mutual information into subterms *does* depend on ψ. Note that the i-th term on the right expresses the mutual information between the i-th bit and the received symbol Y, given the previous $(i-1)$ bits. This can be given an operational meaning which one can use to design a transmission system.

To be concrete, consider the following specific example with $k=2$ shown in Fig. **??**. In this case the above formula reads

$$I(X^1,X^2;Y) = I(X^1;Y) + I(X^2;Y|X^1).$$

The first term can be interpreted as the mutual information between bit X^1 and the received symbol Y considering channel input X^2 as noise, i.e., as part of the channel. More precisely, we have

$$\begin{aligned}
p(y|x^1) &= \frac{p(x^1,y)}{p(x^1)} \\
&= \frac{\sum_{x^2} p(x^1,x^2,y)}{p(x^1)} \\
&= \frac{\sum_{x^2} p(x^1,x^2)p(y|x^1,x^2)}{p(x^1)} \\
&= \sum_{x^2} p(y|x^1,x^2)p(x^2|x^1) \\
&= \sum_{x^2} p(y|x^1,x^2)p(x^2) \\
&= \frac{1}{2}\left(p(y|x^1,x^2=0) + p(y|x^1,x^2=1)\right)
\end{aligned}$$

This transition probability depends on the map ψ. For the two choices of ψ shown in Fig. **??** the resulting transition probabilities are shown in Fig. **??**.

The second term has a similar interpretation, except that now at the receiver we have *side information* X^1, i.e., the term can be interpreted as the mutual information between bit X^2 and the channel output Y given that X^1 is available at the receiver.

Now note that both maps shown in Fig. **??** lead to the same overall capacity but that this mutual information is *split* in different ways between these two *subchannels*. This is true in general.

The abvove interpretation gives rise to the following general *multilevel* scheme. For a given constellation of size 2^k, choose a mapping ψ. Now this gives rise to k channels, where the i-th channel has capacity $I(X^i;Y|X^1,\ldots,X^{i-1})$.

3.3 Bit Interleaved Coded Modulation

One point in the above multilevel scheme that may raise some concern is the dependence of the decision of bit X^i on the previous decisions. This has two

consequences. First, once an error is made at level i it is likely that this error will adversely affect all following levels. This is called *error propagation*. Therefore, in order to limit error propagation, one has to ensure that levels are decoded highly reliably. This usually means large latency.

As we will see now, both these issues can be usually circumvented at a small cost in transmission rate by using a bit-interleaved coded modulation scheme. The basic idea of BICM is straightforward. Since, conditioning increases mutual information we have

$$\begin{aligned} I(S;Y) &= I(X^1,\ldots,X^k;Y) \\ &= \sum_{i=1}^{k} I(X^i;Y|X_1,\ldots,X^{i-1}) \\ &\geq \sum_{i=1}^{k} I(X^i;Y). \end{aligned}$$

The interpretation of the above inequality is immediate. Rather than first decoding bit X^1 and then using this information as side information for decoding bit X^2 and so on, decode all bits in *parallel*. This obviously avoids the latency and error propagation problems. On the other hand, each term $I(X^i;Y)$ is now, in general, *stricly* less than the corresponding term $I(X^i;Y|X^1,\ldots,X^{i-1})$, i.e., the overall transmission rate achievable by BICM is, in general, stricly less than the optimal multilevel scheme. How much is lost now crucially depends on the mapping ψ! In general, a good choice of the mapping ψ is given by the so called *Grey* mapping. This answers the question posed in the beginning. The optimal BICM mapping ψ is the one which maximizes the sum $\sum_{i=1}^{k} I(X^i;Y)$. It turns out that for those constellations most frequently used surprisingly little is lost by employing BICM as opposed to the quite more complicated multilevel scheme!

3.4 ITERATIVE DECODING

It is possible to combine the benefits of both of the above schemes, the high achievable rates of multilevel coding with the simplicity of BICM. This can be done by using iterative schemes. We will have to postpone our discussion of such a scheme until we have discussed the general framework of iterative signal processing.

4. MULTIPLE-ACCESS CHANNEL

Start with a single-user channel and say that we can split the power into two parts and that the combined rate is the same. But since there is no coordination necessary between these two parts we can think of the two transmissions as two separate users. Say that clearly we can not do any better than the obvious bounds from the

single-user theorem. This gives rise to capacity but also rise to an efficient scheme which looks like the multilevel scheme. Now discuss the two user case in more detail.

Compare this with the capacity we can achieve with spread spectrum which is basically a single-user decoding scheme and which is the equivalent of bit interleaving.

5. TRANSMISSION SCHEMES FOR COLORED NOISE: OFDM

5.1 CONSTRAINED OPTIMIZATION: LAGRANGE MULTIPLIERS

Assume we want to optimize the function $f(x,y)$, $f(x,y) \in \mathbb{R}$, under the *constraint* that $g(x,y) = 0$. Assume that at (x_0,y_0) we have a relative extremum of f under the constraint g. Then we must have

$$\begin{aligned} g_x(x_0,y_0)dx + g_y(x_0,y_0)dy &= 0, && (4.1)\\ f_x(x_0,y_0)dx + f_y(x_0,y_0)dy &= 0. && (4.2)\\ & && (4.3)\end{aligned}$$

These equations should be interpreted as follows: Equation (4.1) expresses a constraint on the allowed direction (dx,dy), i.e., we can e.g. fix dx and then express dy as a function of dx. For this we have to assume that at least one of $g_x(x_0,y_0)$ or $g_y(x_0,y_0)$ is unequal to zero. Equation (4.2) on the other hand expresses the usual condition that the function f has a stationary point at (x_0,y_0) (with respect to the allowed direction). In this sense (4.2) does *not* impose a further restriction on the allowed direction. Consider now the matrix

$$\begin{pmatrix} f_x(x_0,y_0) & f_y(x_0,y_0) \\ g_x(x_0,y_0) & g_y(x_0,y_0) \end{pmatrix}$$

We claim that this matrix has rank exactly one. To see this, note that it can not have rank zero by our assumption that at least one of $g_x(x_0,y_0)$ or $g_y(x_0,y_0)$ is unequal to zero. Further, the matrix can not have rank two since otherwise (4.2) would impose another restriction on the allowed direction, contrary to our assumption.

The rank deficiency of the above matrix implies that there must be a constant, call it λ such that

$$\begin{aligned} f_x(x_0,y_0) + \lambda g_x(x_0,y_0) &= 0,\\ f_y(x_0,y_0) + \lambda g_y(x_0,y_0) &= 0. \end{aligned}$$

and we also still have the constraint $g(x_0,y_0) = 0$. The factor λ is called the *Lagrange multiplier*.

In summary, in order to perform the desired constrained optimization consider the *Lagrangian*

$$L(x,y,\lambda) = f(x,y) + \lambda g(x,y)$$

and look for stationary points of $L(x,y,\lambda)$. This leads exactly to the desired set of equations.

5.2 PARALLEL GAUSSIAN CHANNELS

Assume we have k *parallel* Gaussian channels, where the i-th channel has power constraint P_i and noise variance σ_i^2. Clearly, then we can transmit over these parallel channels at least at rate

$$\sum_{i=1}^{k} \frac{1}{2} \log(1 + \frac{P_i}{\sigma_i^2}), \tag{4.4}$$

and, converesly one can show that this is indeed the maximum rate. Assume now that we have *total* power P and that we can split up this power into k parts in any desired way in order to maximize the resulting sum rate.

Therefore, we want to maximize (4.4) under the constraints

$$\begin{aligned} P_i &\geq 0, \\ \sum_{i=1}^{k} P_i &= P. \end{aligned}$$

If, for a second we ignore the non-negativity constraints on the P_i then from the above section we know that in order to find stationary points we have to look at the Lagrangian

$$\sum_{i=1}^{k} \frac{1}{2} \log(1 + \frac{P_i}{\sigma_i^2}) + \lambda \sum_{i=1}^{k} P_i$$

Taking the derivatives with respect to each P_i we find that

$$P_i = \nu - \sigma_i^2.$$

We claim that if we include our non-negativity constraints then the optimial solution is given by

$$P_i = (\nu - \sigma_i^2)^+,$$

where ν is chosen so that $\sum_{i=1}^{k} (\nu - \sigma_i^2)^+ = P$. This can be verified as follows. Note that

$$(\nu - \sigma_i^2)^+ = \begin{cases} \nu - \sigma_i^2, & \nu \geq \sigma_i^2, \\ 0, & \text{otherwise.} \end{cases}$$

Assume now that we perturb our solution by a vector $(dP_1, \ldots, dP_k)$, where we must have $dP_i \geq 0$ for all those i such that $\nu \leq \sigma_i^2$ and $\sum_{i=1}^k dP_i = 0$. The change in our function is then equal to

$$\frac{1}{2} \sum_{i=1}^{k} \frac{dP_i}{(\nu - \sigma_i^2)^+ + \sigma_i^2}.$$

If we can show that for any vector $(dP_1, \ldots, dP_k)$ which fulfills the constraint the change in capacity is non-positive then we have shown that our proposed solution is at least a local maximum. But note that $\frac{1}{(\nu-\sigma_i^2)^+ + \sigma_i^2} = \frac{1}{\nu}$ if $\nu \geq \sigma_i^2$ and is equal to $\frac{1}{\sigma_i^2} \leq \frac{1}{\nu}$ otherwise. This shows that the proposed point is at least locally optimal. The above solution has a nice gemetric interpretation which is known as *waterpouring.*

5.3 General Channel With Colored Noise

From the above considerations we can give a (heuristic) derivation of the capacity of a Gaussian noise channel with colored noise. Consider the model

$$Y(t) = X(t) + Z(t)$$

where we have again power constraint P and where $Z(t)$ is a wide-sense stationary Gaussian process with power-spectral density equal to $N(f)$.

Consider the channel and split the frequency axis into many small "slices". Consider one such slice centered at frequency f_i and of width ΔW. Assuming that $N(f)$ various sufficiently slowly around f_i this channel has approximately constant power spectral density along its region of interest, and this constant is equal to $N(f_i)$. Assume that we assign power P_i to this slice. Then, from our previous results we know that the capacity of this slice is equal to

$$\Delta W \ln \left(1 + \frac{P_i}{2N(f_i)\Delta W} \right),$$

where the factor two appears since we assume that $N(f)$ is the two-sided power spectral density.

Proceeding in the same way for all slices we see that the total rate we can achieve with such a scheme is equal to

$$\sum_i \Delta W \ln \left(1 + \frac{P_i}{2N(f_i)\Delta W} \right)$$

where we must have $\sum_i P_i = P$. If we now let the number of slices tend to infinity and define $P(f)$ as the limit of $P_i/(2W)$ then we see that the achievable rate for such a scheme is

$$\int_{-\infty}^{\infty} \frac{1}{2} \ln \left(1 + \frac{P(f)}{N(f)} \right) df$$

As discussed in the previous section, we will achieve maximal sum rate if we perform water filling in the spectrum. Therefore, for the optimal power allocation the maximally achievable sum rate is equal to

$$\int_{-\infty}^{\infty} \frac{1}{2} \ln \left(1 + \frac{(\nu - N(f))^+}{N(f)} \right) df$$

where ν is chosen such that

$$\int_{-\infty}^{\infty} (\nu - N(f))^+ df = P.$$

Note that the above derivation is quite heuristic but essentially correct. For a more rigorous derivation consult your information theory book.

5.4 OFDM

The above derivation gives rise to the following transmission scheme which is called orthogonal frequency division modulation. Assume that we mimick in the transmission process the derivation from above, i.e., we slice the spectrum into many small parts and essentially transmit over each such slice separately. The advantage of such a scheme is that over the (small) bandwidth of each slice the noise spectrum is more or less constant and therefore we do not need sophisticated equalization techniques to approach the capacity of this slice. On the other hand there are also the following two disatvantages. First, since the bandwidth of such a slice is very small the corresponding symbol rate is very small and therefore on each channel we signal at a very low rate which implies large delays. Second, at the transmitter we transmit the sum of many (more or less) independet random signals. This implies that we have a large *peak-to-average* power ratio. In particular, assume that P is the average power and that it is split into N parts, each transmitting at power P/N. To be specific assume that each signal is simply a complex sinuisoid of the form $\sqrt{\frac{P}{N}} e^{2\pi j f_0 t}$. If at any point in time all signals align then we get a peak power equal to $(N\sqrt{\frac{P}{N}})^2$ compared to the average power of P. Therefore the peak power can be a factor N larger than the average power which can cause significant problems in actual systems.

SYSTEM DESCRIPTION

Consider now a specific system. Assume we split the band into equal N width slices each of bandwidth ΔW. Let the center frequency in the i-th band be equal to $\Delta f_0 i$. Assume that in the ith band we use standard pulse-amplitude modulation where we choose the basic signal constellation in such a way that we can transmit close to capacity over this band. Let T denote the length of one symbol interval. Consider now the transmitted signal in one such symbol interval. For simplicity

of notation we will assume that this interval is $[0,T)$. The complex baseband equivalent signal for the i-th frequency slot in this interval has the form

$$s_i(t) := x_i e^{2\pi j f_0 (i-\frac{N-1}{2})t}, \ 0 \le t < T,$$

where x represents the information and the shift $\frac{N-1}{2}$ ensures that the overall signal occupies the minimum bandwidth in baseband. The modulating signal x_i is complex valued and typically is an element of one of the standard signaling sets (PSK, QAM, ...). The overall signal in one basic symbol interval is therefore

$$s(t) = \sum_{i=0}^{N-1} s_i(t) = \sum_{i=0}^{N-1} x_i e^{2\pi j f_0 (i-\frac{N-1}{2})t}, \ 0 \le t < T.$$

We first consider the question how we should choose the frequency separation f_0 with respect to the symbol interval T. Consider the i-th and the k-th signal. We claim that these two signals will be orthogonal if we choose $f_0 = \frac{1}{T}$. This follows since

$$\begin{aligned} < s_i(t), s_k(t) > &= \int_0^T x_i e^{2\pi j f_0 (i-\frac{N-1}{2})t} x_k^* e^{-2\pi j f_0 (k-\frac{N-1}{2})t} dt \\ &= \int_0^T x_i e^{2\pi j f_0 it} x_k^* e^{-2\pi j f_0 kt} dt \\ &= x_i x_k^* \int_0^T e^{\frac{2\pi j (i-k)t}{T}} dt \\ &= \begin{cases} 0, & i \neq k, \\ |x_i|^2, & i = k. \end{cases} \end{aligned}$$

Note: As you will see in Exercise 4.4 this is actually not the densest possible spacing but it ensures that the phases are continuous, and therefore this is in practice the preferred method. With the above choice $f_0 = \frac{1}{T}$ the signal takes on the form

$$s(t) = \sum_{i=0}^{N-1} x_i e^{\frac{2\pi j (i-\frac{N-1}{2})t}{T}} = e^{\frac{2\pi j (N-1)t}{2T}} \sum_{i=0}^{N-1} x_i e^{\frac{2\pi j i t}{T}}, \ 0 \le t < T. \qquad (4.5)$$

EFFICIENT IMPLEMENTATION

At first OFDM might seem a costly approach to transmitting the signal. Typically, in a receiver the physical components like modulator (which shifts the signal in the transmission band) and the power amplifier are the most costly components. For OFDM it seems we need as many as N modulators, and for proposed systems this N can be as high as 1024. Fortunately this is not the case. Consider again the OFDM signal as given in equation (4.5). Then we can write it as

$$s(t) = e^{\frac{2\pi j (N-1)t}{2T}} \sum_{i=0}^{N-1} x_i e^{\frac{2\pi j i t}{NT_s}}, \ 0 \le t < T.$$

where $T_s := \frac{T}{N}$. The right hand side of the equation should now look familiar. It is basically the equation for the discrete fourier transform and it can therefore be evaluated efficiently, in particular so if N is chosen to be a power of two, which is virtually always the case in practice.

Ignoring the common factor $e^{\frac{2\pi j(N-1)t}{2T}}$ we therefore get

$$s(kT_s) \sim \sum_{i=0}^{N-1} x_i e^{\frac{2\pi jik}{N}},$$

so that we recognize T_s as *sampling* interval. Is this sampling rate sufficient? Note that the highest frequency of the signal is of order $f_0\frac{N}{2}$. By Nyquist we have to take samples at most $\frac{1}{2f_0\frac{N}{2}} = \frac{T}{N} = T_s$ apart! This analysis somewhat ignores the effect of the modulation but it is correct to first order. If we wish to obtain a higher sampling rate, we can always append a sufficient number of zeros to the frequency signal and take a fourier transform of larger length at the cost of increased complexity.

Note that the signal-to-noise ratio can be quite different in each subband. Therefore, we first have to decide how much of our power we want to allocate for each frequency slot. As a guideline we can use our waterpouring approach. Next, given the power allocation we have to decide what modulation we should use in each band. Again, we can simply use our guidlines from the single carrier case.

PEAK-TO-AVERAGE POWER RATIO

As discussed in the introduction, one practical disatvantage in a multicarrier system is its inherent high signal-to-noise power ratio. This stems from the fact that we chose the signal to be composed of a sum of orthogonal signals, so that the average power is simply the sum of the average powers of the subsignals but the peak can (and typically is!) a factor N larger. This causes problems with amplifiers as well as with regulations in which a maximum peak power is prescribed.

There are several approaches to limit deal with this problem and we will discuss them now briefly.

The first approach is to restrict the set of N-tuples $(x_0, \ldots, x_{N-1})$ to those which give a low PAPR. To be precise, assume that all x_i live in the same alphabet, e.g. $\{\pm 1\}$. We can then restric the set of all 2^N N-tuples to those which have a PAPR of at most c, $c \geq 1$. If we choose e.g. $c = 2$, then there is a well studied set of such sequences, called *Golay* complementary sequences, see Exercise 4.5. There are many problems associated with this approach. First, finding the set of admissable N-tuples is highly non-trivial, and describing them in a compact way is even more difficult. This is even more true as in general we want to choose the components x_i from different alphabets and these choices might vary with time as the channel conditions vary.

There are more probabilistic approaches to the problem. First note that it is known that for large N, "most" N-tuples have a PAPR of approximately $\ln(N)$. Therefore, if $\ln(N)$ is acceptable then most N-tuples do not violate our constraint. The violation of the PAPR constraint is therefore a rare event. This means the following: The cardinality of the set of sequences which PAPR below $\alpha \ln(N)$ is roughly speaking equal to the whole space, and so we do not need to decrease our rate substantially. How can we weed out now the few bad N-tuples. The simplest approach is to simply *clip* the transmitted signal. This will add additional "noise" to the system which we can counteract by an appropriate choice coding. A more structure approach is to keep a few *spare* frequencies distributed in the spectrum. For every tuple we now choose these spare frequencies in such a way that they minimize the PAPR. This reduces our rate by the ratio of spare frequencies and it requires us to solve the optimization problem. Finally, we can introduce at the transmitter intentionally a small number of "errors" so that the resulting signal again fulfills the PAPR requirement. Again, we counteract the effect of these errors by a proper coding scheme. The last three approaches (or a combination of them are the most practical).

THE CYCLIC PREFIX TRICK

Recall that the whole motivation for OFDM was the fact that within each narrow frequency band the channel was approximately constant so that we need no or only simple equalizers. Nevertheless, in order to use the FFT at the encoder and decoder we still have to deal with the channel impulse response.

In more detail. Consider the signal $s(t)$ in one symbol interval. Assume that the signal is sent through a channel with impulse response $h(t)$ and that we add AWGN with double sided power spectral density equal to N_0 (everything here is complex valued). Since we are looking at the complex baseband equivalent model which has bandwidth approximately equal to $\frac{N}{2T}$ (ignoring the effect of the modulation signal and the effect of filtering) we can at the receiver first perform a low pass filtering with the equivalent bandwidth to reject out of band noise and then sample the signal with samples taken roughly $\frac{T}{N}$ second apart. The effect of this operation is that we are now dealing with a discrete time model

$$s_n = \sum_k h_k s_{n-k} + z_n, \quad , i = 0, \ldots, N-1.$$

We now like to take the invers fourier transform (which is an orthogonal transform and therefore keeps the noise invariant) to get back to the (noisy) versions of the transmitted symbols. The problem with this transition occurs at the boundaries. Adjacent symbols will "bleed" into the interval under consideration and cause intersymbol interference. The standard approach to dealing with this problem is to keep "guard" intervals between adjacent symbol intervals and to send in these guard intervals a cyclic extension of the signal whose size is at least equal to the length of channel response. Since the cyclic convolution of two signals of length

N is equal to the multiplication of their respective (length N) signals, this trick eliminates intersymbol interference, although at the cost of a decrease in rate.

EXERCISES

4.1. In class we considered the capacity region for the two-user multiple-access channel. Trusting your intuition, write down the set of inequalities defining the capacity region for the three-user case. You can do this either in terms of (conditional) mutual informations or for the specific case of the Gaussian multiple-access channel. Can you sketch a "typical" such region. (In two dimensions this region was a pentagon.) Is there again a similar interpretation of the "corner points."

4.2. Consider now the capacity region of a k-user multiple access channel and assume (as we always have done) that all users are *synchronized*, i.e., they have access to a common clock. Argue that in this case the capacity region has to be convex, i.e., if $R = (R_1, \ldots, R_k)$ and $R' = (R'_1, \ldots, R'_k)$ are achievable rate tuples then for any $\alpha \in [0,1]$ also $\alpha R + (1-\alpha)R'$ is achievable.

4.3. For part of this exercise you will need computer access. You can form groups and if you hand in the solution to this exercise you will get extra credit. If you do not have computer access but want to do this exercise let us know. It might be handy to know that in *Mathematica* you can use the following commands: $NIntegrage[f[x], \{x, xmin, xmax\}]$ gives you the (numerical) integral of the function $f[x]$.

Consider 4-PAM mudulation with the points $\{-3,-1,1,3\}$ and the following two labeling mappings. The first is the Gray mapping where the points have consecutive labels $\{00,01,11,10\}$, the second is the mapping with consecutive labels $\{00,01,10,11\}$. Assume that the channel is Gaussian with variance $\sigma^2 = \frac{1}{2}$. For both maps find the maximum achievable rate under multilevel coding and under bit-interleaved coded modulation (BICM). Which map is perferrable for BICM.

4.4. Consider the complex baseband equivalent OFDM signal $s(t) = \sum_{i=0}^{N-1} s_i(t)$, $t \in [0,T)$, in one symbol interval as discussed in class. What is the corresponding passband signal? Next assume that we have coherent detection. This means that at the receiver we can separate the inphase and the quadrature components of the signal. Argue now that we could have chosen the spacing of the frequencies twice as dense, namely we could have chosen $f_0 = \frac{1}{2T}$ and still maintained the orthogonality relationship between the signals.

4.5. In this exercise we will learn some simple facts about binary Golay sequences. Consider two binary sequences x and y of length N, more precisely $x,y \in \{\pm 1\}^N$. We say that x and y are *complementary*, which we denote by $x \sim y$ if

$$\sum_k (x_k x_{k+i} + y_k y_{k+i}) = 2N\delta_i,\ i \in \mathbb{Z}, \tag{4.6}$$

where we assume here that x and y are non-zero for $k = 0, \ldots, N-1$, and are zero outside this region.

Example 14. The simplest exmaple of a complementary pair is $++ \sim +-$. Examples of complementary pairs of length 4, 8, and 10 are $+++- \sim ++-+$, $+++-++-+ \sim +++---+-$, and $-++-+---- \sim +-+---++--$.

For a given $x \in \{\pm 1\}$, define let its discrete-time Fourier transform be denoted by $X(e^{2\pi jf})$,

$$X(e^{2\pi jf}) = \sum_k x_k e^{2\pi jfk}.$$

Use Parseval's theorem to show that

$$\max_{f\in[0,1)} |X(e^{2\pi jf})|^2 \geq N.$$

Next, translate the relationship in (4.6) into the frequency domain and show that it reads

$$|X(e^{2\pi jf})| + |Y(e^{2\pi jf})| = 2N, \ f \in [0,1).$$

Now argue that this shows that the PAPR for an N-tuple which is complementary is at most 2. Finally, can you prove any of the following *generation* rules which generate longer Golay pairs from shorter ones? In the following let $x \sim y$. Let $-x$ denote the sequence all of whose components are negated, and let $\hat{x}$ denote the "time-reversed" sequence.

- $\hat{\pm}x \sim \hat{\pm}y$
- $a \sim b$, where $a_i = (-1)^i x_i$ and $b_i = (-1)^i y_i$
- $xy \sim x(-y)$
- $x_0 y_0 x_1 \ldots x_{N-1} y_{N-1} \sim x_0(-y_0)x_1 \ldots x_{N-1}(-y_{N-1})$

5

A GLIMPSE AT ITERATIVE CODING

1. INTRODUCTION

Information theory establishes the limits of communications–what is achievable and what is not. *Coding theory* tries to devise low complexity schemes that approach these limits.

The general problem of point-to-point communications is to *convey a source reliably to a sink over a given channel* as shown in Fig. 5.1. The *source* might

Figure 5.1: The basic point-to-point communications problem.

refer to a picture, a sound pattern or a piece of text. Important examples of *channels* which one encounters in everyday life are radio links, phone lines or fiber optic cables. For these examples, information is transmitted from *one point in space to another*. But there are also important examples for which information is transmitted from one *point in time to another*. The most familiar examples are probably magnetic storage systems and compact discs. The *sink* serves simply as a reminder that we would like to reconstruct the transmitted information given the channel output with high reliability.

2. SHANNON'S FRAMEWORK

In his seminal paper in 1948 Shannon showed that without loss of generality the point-to-point problem can be broken down into two separate problems as shown

in Fig. 5.2. First, represent the source as efficiently as possible given a desired upper bound on the distortion . This process is called *source coding* . Shannon's *source coding theorem* asserts that for a given source there exists a minimum rate $R = R(d)$ which is necessary (and sufficient) to describe this source with distortion not exceeding d. The plot of this required rate R as a function of the distortion d is usually called the *rate-distortion* curve. In the second stage an appropriate amount of redundancy is added to these source bits to protect them against the errors in the channel. This process is called *channel coding* . Shannon's *channel coding theorem* asserts that there is a maximum rate at which information can be transmitted reliably, i.e., with vanishing probability of error, over a given channel. This maximum rate is called the *capacity* of the channel. At the receiver we first decode the received bits in order to determine the transmitted information. We then use the decoded bits to reconstruct the source at the receiver. Shannon's *source-channel separation* theorem now asserts that the source can be reconstructed with a distortion of at most d at the receiver as long as $R(d) < C$, i.e., as long as the rate required to represent the given source with the allowed distortion is smaller than the capacity of the channel. Further, no scheme can do better. In this section we

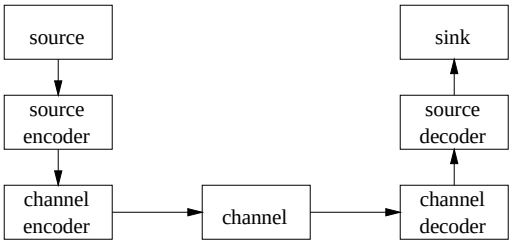


Figure 5.2: The basic point-to-point communications problem.

will not be concerned with the source coding problem. We will assume that the source emits a sequence of i.i.d. bits which are equally like zero or one. Under this assumption we will see how to accomplish the channel coding problem in an efficient manner for a variety of scenarios.

3. IMPORTANT CHANNEL MODELS

Let x be the input and y be the output of a given channel $p(y|x)$, where x and y are both of length n. We will only be concerned with *memoryless* channels, i.e., channels for which $p(y|x) := \prod_{i=1}^{n} p(y_i|x_i)$. Further, we will only deal with *binary-input* channels, i.e., channels with an input alphabet I of cardinality two. In all our cases the input alphabet will either be $\{0,1\}$ or $\{\pm 1\}$. Let α and $\overline{\alpha}$ denote the two possible inputs to a binary-input channel. We say that such a channel is *output-symmetric* if $p(-y|\alpha) = p(y|\overline{\alpha})$. We note that the binary-input channels which are most important in practice have this property.

In particular we will often use the following three binary-input output-symmetric memoryless channels as examples. These are the binary erasure channel (BEC), the binary symmetric channel (BSC), and the binary-input additive white Gaussian noise channel (BIAWGNC). For completeness we review the basic facts concerning these three channel.

Example 15. [BEC] Fig. 5.3 shows the binary erasure channel. Every transmitted bit is either *erased* with probability ϵ or otherwise transmitted correctly. The random variables which determined whether bits are erased or not are independent. The capacity of this channel is $C_{\text{BEC}} = 1 - \epsilon$ bits per channel use. The BEC can be used as a naive first model to model packet losses due either to buffer overflows or to excessive delays in a packet network (if we assume that packet losses are independent of each other). The fact that any bit which is not erased is known to be *correct* generally facilitates the analysis of any coding system for the BEC considerably and coding for this channel is far better understood than for any other non-trivial channel. □

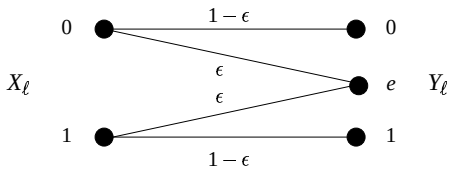


Figure 5.3: The binary erasure channel (BEC) with erasure probability ϵ.

Example 16. [BSC] Fig. 5.4 shows the binary symmetric channel. Every transmitted bit is either *flipped* with probability ϵ or otherwise transmitted correctly. The random variables which determined whether bits are flipped or not are independent. The capacity of this channel is $C_{\text{BSC}} = 1 - h(\epsilon)$ bits per channel use. The BSC is the generic model for any memoryless channel in which hard decisions are performed at the front end of the receiver. □

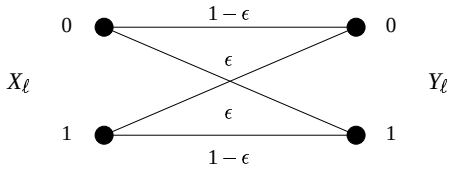


Figure 5.4: The binary symmetric channel (BSC) with cross-over probability ϵ.

Example 17. [BIAWGNC] Fig. 5.5 shows the binary-input additive white Gaussian noise channel. The input alphabet of the channel is $\{\pm 1\}$. Denote the channel

input at time ℓ by X_ℓ, $X_\ell \in \{\pm 1\}$, and the channel output by Y_ℓ, $Y_\ell \in \mathbb{R}$. For the BIAWGNC we have $Y_\ell = X_\ell + Z_\ell$, where Z_ℓ is a normal random variable with zero mean and variance σ^2, and where the sequence of random variables $\{Z_\ell\}_\ell$ is independent. The capacity of this channel is

$$C_{\text{BIAWGNC}} := -\int_{-\infty}^{\infty} \phi(x) \log_2 \phi(x)\, dx - \frac{1}{2}\log_2 2\pi e\sigma^2 \text{ bits per channel use,}$$

where $\phi(x) = \frac{1}{\sqrt{8\pi\sigma^2}}\left[e^{-\frac{(x-1)^2}{2\sigma^2}} + e^{-\frac{(x+1)^2}{2\sigma^2}}\right]$. Note that the capacity is a function of

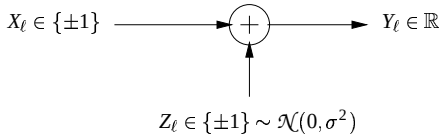


Figure 5.5: The binary-input additive white Gaussian noise channel with noise variance σ^2.

$1/\sigma^2$ alone. More generally, if we allow a scaling of the inputs then the capacity is a function of $\frac{E_N}{\sigma^2}$, where E_N is the energy expanded per channel use (dimension). A plot of C_{BIAWGNC} as a function of E_N/σ^2 is shown in Fig. 5.6. Also shown is

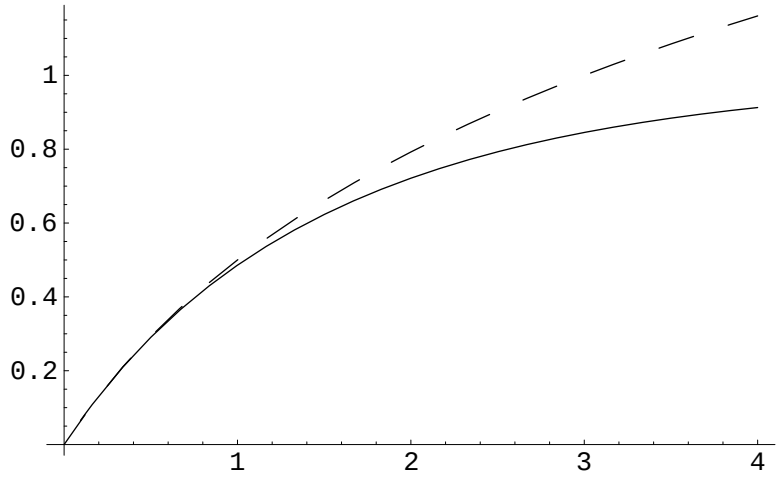


Figure 5.6: Capacity of the BIAWGNC (solid line) and the AWGNC (dashed line) as a function of $\frac{E_N}{\sigma^2}$.

the capacity of the regular additive white Gaussian noise channel (AWGNC) with real-valued inputs which is equal to $C_{\text{AWGNC}} := \frac{1}{2}\log_2(1 + \frac{E_N}{\sigma^2})$ bits per channel use.

Clearly, for small rates (small values of E_N/σ^2) we pay only a small penalty for restricting the input to be binary. In order to assess the performance of a code over the BIAWGN channel it is natural to plot the bit error probability P_b as a function of $\frac{E_N}{\sigma^2}$. For small rates r it is even more useful to plot the bit error probability as a function of $\frac{E_b}{N_0}$, where $E_b = \frac{1}{r}E_N$ is the energy expanded per bit and $N_0 = 2\sigma^2$ is the one-sided power spectral density so that $\frac{E_b}{N_0} = \frac{1}{2r}\frac{E_N}{\sigma^2}$. Why is it convenient to use $\frac{E_b}{N_0}$ for small rates?

In this case a quick calculation shows that $C_{\text{BIAWGNC}} \sim C_{\text{AWGNC}} \sim \frac{E_N}{2\sigma^2}$ bits per channel use (just use the Taylor series expansion on the expression for C_{AWGNC}). Assume we transmit at rate $r = \alpha C_{\text{AWGNC}}$, i.e., we achieve a fraction α of capacity. In this case we see from the approximation that we have $\frac{E_b}{N_0} = \frac{1}{\alpha}$, a constant! In other words, for low rates measuring the performance with respect to $\frac{E_b}{N_0}$ allows us to compare codes of different rates on an (almost) equal footing.

Let C denote the Shannon capacity of a given channel. Then any rate below C can be achieved with vanishing probability of error and vice-versa to achieve a vanishing probability of error we have to transmit below C. What if we allow a non-vanishing probability of error, lets say p? What is then the maximal rate at which we can transmit? Call this rate $C^{(p)}$. In this case we can proceed as follows: First compress the information such that the original bits can be reconstructed from the compressed version with a Hamming distortion of (at most) p. From elementary rate-distortion theory we know that this requires a source code of rate $1-h(p)$. These compressed bits can now be transmitted over the channel at vanishing probability of error, so that the condition for successful transmission reads $r(1-h(p)) < C$. Further, by the source-channel separation theorem for point-to-point channels this is the best we can do. It follows that $C^{(p)} = \frac{C}{1-h(p)}$. To be concrete, consider a channel parametrized by a single real valued parameter x and with capacity $C(x)$ so that $C(x)$ is a strictly increasing function in x (e.g. the BIAWGNC parametrized by E_b/N_0). From our remarks above we see that in order to transmit over this channel at rate r with a bit error probability of at most P_b requires that $r < \frac{C(x)}{1-h(P_b)}$ or reversely that $x > C^{-1}(r(1-h(p))$. This is demonstrated for $r = \frac{1}{2}$ in the case of the BIAWGNC in Fig. 5.7. □

At first it might seem that these three channel models hardly begin to scratch the surface of the rich class of channels that one might encounter in practice and that, therefore, our focus on these three models might limit the applicability of our results to a very narrow range of applications. Fortunately, the situation is not quite as drastic. First, these three models appear unusually often in practice. Second, it is at least in theory possible to build up more complex models by using these three channel models as building blocks. E.g., once the BIAWGNC is mastered the general AWGNC could simply be handled by *stacking-up* properly scaled codes for the BIAWGNC. Third, the extension of many of the methods and theorems which we will discuss to more general scenarios is often quite straightforward and

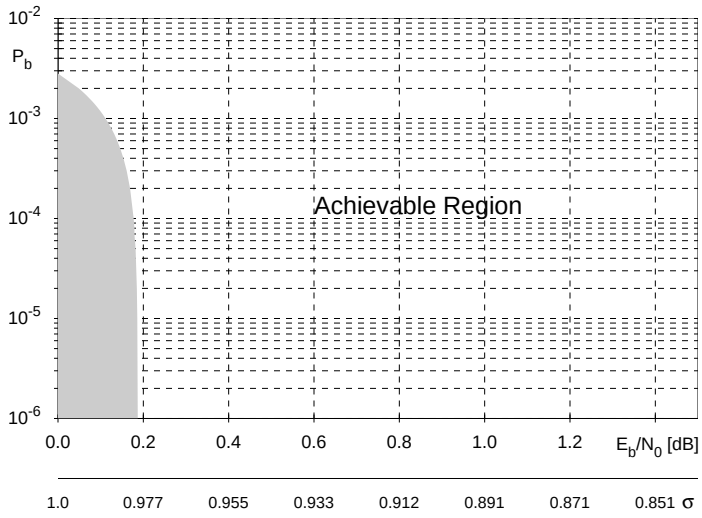


Figure 5.7: The achievable region for non-vanishing P_b and the BIAWGNC.

so for the sake of notational simplicity we will not discuss them in detail.

4. CODING: (TWO) TRIAL(S, THEIR RATE) AND (THEIR ASSOCIATED) ERROR

How can we transmit information reliably over a noisy channel at a positive rate? Consider the following two ad-hoc schemes for transmission of information over the BSC with cross-over probability ϵ, where without loss of generality we can assume $\epsilon \leq \frac{1}{2}$.

First consider *uncoded* transmission, i.e., we simply send the information across the channel without the insertion of redundant bits. Let x, $x \in \{0,1\}$, denote the desired bit which we would like to convey to the receiver and let y denote the received bit. The MAP estimator chooses $\hat{x} := \text{argmax}_{x\in\{0,1\}} P(x|y)$. Since the bits are equally likely and since $\epsilon \leq 1/2$ this estimate is equal to $\hat{x} = y$ and therefore the bit error probability is equal to $P_b = \text{P}\{\hat{x} \neq x\} = \epsilon$. We can therefore achieve with this transmission strategy a (rate, P_b)-pair of $(1, \epsilon)$.

If this error probability is too high, what transmission strategy can we use to lower the error probability? The simplest such strategy is *repetition-coding*. Assume we repeat each bit k times, where for simplicity we will assume that k is odd. So if we want to transmit bit x then the input to the BSC is $\underbrace{x \cdots x}_{k\times}$. Denote

the k received bits by $y_1 \cdots y_k$. It is easy to check that the MAP decoding rule is $\hat{x} =$ majority of $\{y_1, \cdots, y_k\}$. Hence the probability of bit error is given by

$$P_b = \mathrm{P}\{\hat{x} \neq x\} = \mathrm{P}\{\lceil \frac{k}{2} \rceil \text{ errors occur}\} = \sum_{i>k/2} \binom{k}{i} \epsilon^i (1-\epsilon)^{k-1}.$$

So with repetition codes one can achieve the pairs $(\frac{1}{k}, \sum_{i>k/2} \binom{k}{i} \epsilon^i (1-\epsilon)^{k-1})$. Clearly, in order to have P_b approach zero one has to choose k larger and larger and as a consequence the rate will approach zero as well!

Can we keep the rate positive while letting the bit error probability go to zero?

There are quite a few different types of coding schemes available. In your first communications class you have encountered *convolutional codes* and in the information theory class you learned about block codes, in particular Hamming codes and Reed-Solomon codes. These are the classical codes which are used extensively (and almost exclusively) in applications. As a rule of thumb, for reasonable complexities and bit error probabilities of roughly 10^-5, convolutional codes allow the transmission of information over Gaussian channels at roughly twice the energy per transmitted symbol as compared to Shannon's bound. (In engineering jargon we say that such codes are roughly 3dB away from capacity.) We will now investigate a completely different approach to coding based on (sub-optimal) iterative decoders. This approach was originally invented by Gallager in the early 60ties. Since at that time the necessary technology to implement these techniques was note available, iterative coding schemes were completely forgotten until the early nineties, when they were rediscovered. It is now known, that for large enough block lengths iterative coding techniques allow transmission of information very close to capacity at extremely low complexities.

In this section we will focus almost exclusively on the simplest channel, the BEC, since for this case the analysis is particularly simple. But the principle of iterative decoding is readily extended to general channels and the best (practical implementable) codes known to date are all based on iterative decoding schemes.

In principle iterative decoding can be applied to any given code. But, unless the code is constructed in a proper way, the resulting performance will be very poor. E.g. Reed-Solomon are not suitable for iterative decoding. Of the many classes of codes which have been designed for iterative decoding (turbo-codes, repeat-accumulate codes, tree codes, low-density parity-check codes, ...) we will focus on low-density parity-check codes. These are the codes originally invented by Gallager.

5. LOW-DENSITY PARITY-CHECK CODES

A binary linear code C of length n and dimension k can be described in terms of its $(n-k)\times n$ *parity-check matrix* H,

$$C(H) := \{x \in \mathrm{GF}(2)^n : Hx^T = 0^T\}.$$

In a nutshell, low-density parity-check (LDPC) codes are linear codes which have *sparse* parity-check matrices (hence also the name *low-density*).

Given a binary linear code we would like to associate to it a graphical representation. For this purpose we will use the following simple bipartite graph: the *variable* or *left* nodes corresponds to components of the codeword and the *check* or *right* nodes correspond to the set of parity-check constraints satisfied by codewords of the code.

Example 18. Assume we are given a code $C(H)$ of length $n = 10$ and dimension $k = 5$, where

$$H := \begin{pmatrix} 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \end{pmatrix}.$$

The bipartite graph representing $C(H)$ is shown in Fig. 5.8. Note that each check node represents one linear constraint (one row of H). For the particular example we start with ten degrees of freedom (ten variable nodes). The five linear constraints reduce the number of degrees of freedom by at most five (and exactly by five if all these linear constraints are linearly independent as is true for this specific example). Therefore at least five degrees of freedom remain. It follows that the shown code has rate (at least) one-half. □

Note that in the above example every variable node has degree three and every check node has degree six – in other words every component participates in three checks and every check involves six components. We call such a code a $(3,6)$-regular *low-density* parity-check code. Why *low-density*? Note that the number of *one* entries in the parity check matrix is exactly $3n$, where n is the length of the code. In particular, if we think of $(3,6)$-regular codes of increasing lengths then the number of one entries in the parity check matrix only grows *linearly* with the length. This is in contrast to lets say a random linear code, where each entry in the parity check matrix is chosen i.i.d. to be one with probability one-half, so that the number of one entries in such a parity check matrix grows like the square of the code length.

More generally, a $(\mathrm{d_l}, \mathrm{d_r})$-regular LDPC code is a binary linear code such that every component participates in exactly $\mathrm{d_l}$ checks and such that every check

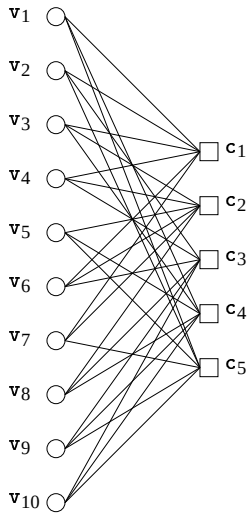


Figure 5.8: A $(3,6)$-regular code of length 10 and rate at least one-half. There are 10 variable nodes and 5 check nodes. For each check node $\mathtt{c}_i$ the sum (over GF(2)) of all adjacent variable nodes is equal to zero.

involves exactly $\mathrm{d_r}$ components. Note that the *rate* of such a code is equal to $r := 1 - \frac{\mathrm{d_l}}{\mathrm{d_r}}$. This can be seen since on the one hand there are $n\,\mathrm{d_l}$ edges emanating from all variable nodes. On the other hand, assuming that there are m check nodes, then this number of edges is also equal to $m\,\mathrm{d_r}$. Equating these two expressions we get $m = n\frac{\mathrm{d_l}}{\mathrm{d_r}}$, from which the rate expressions follows.

6. ITERATIVE DECODING OF LDPC CODES FOR THE BINARY ERASURE CHANNEL

We will restrict our discussion to the simplest channel, namely the binary erasure channel (BEC). For this channel iterative decoding is particularly simple. Consider the code with a graphical representation as given in Fig. 5.9. In the left picture a particular codeword is marked. All edges emanating from variables nodes which are one are drawn solid whereas those which emanate from variable nodes which are zero are drawn dashed. It is easy to verify that this indeed constitutes a valid codeword by checking that every check node has an *even* number of *solid* edges emanating from it. On the right the same codeword is shown after it is passed through a BEC. Seven of the bits have been erased and they are marked by "?".

The workings of the iterative decoding procedure which we consider are

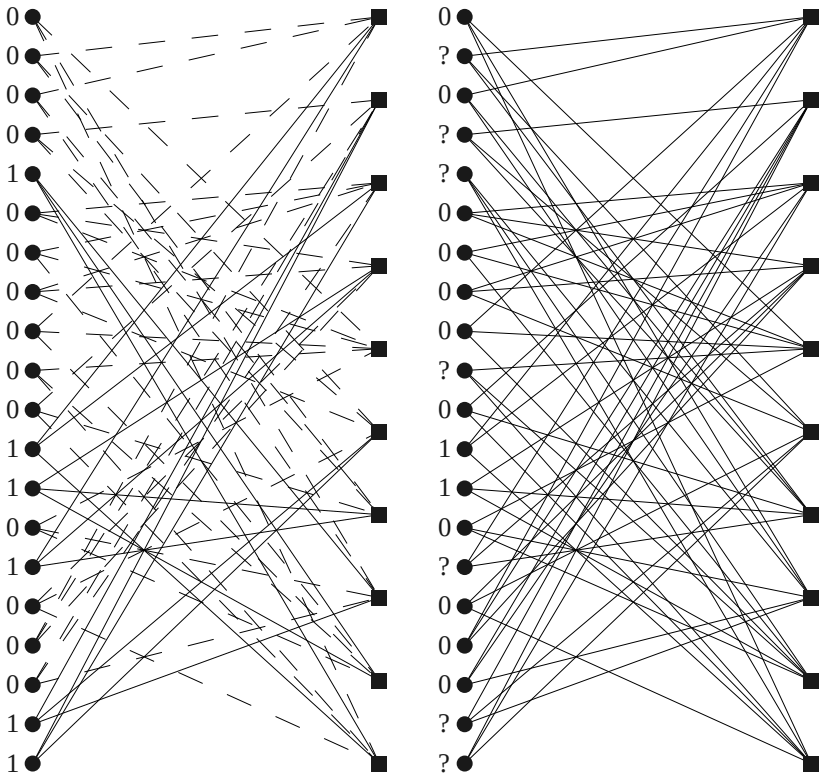


Figure 5.9: A $(3,6)$-regular LDPC code of length 20. In the left figure a particular codeword is shown. All edges emanating from variables nodes which are one are drawn solid whereas those which emanate from variable nodes which are zero are drawn dashed. On the right the same codeword is shown after it is passed through a BEC. Seven of the bits have been erased and they are marked by "?".

shown in Fig. 5.10. As a first step, each variable node which is not erased propagates its known value along all its outgoing edges. This is shown in the left column of the figure which is entitled `left-to-right`. Edges which carry a zero are drawn as thin lines whereas edges which carry a one are shown as thick lines. Edges which emanate from erasures are drawn as dashed lines. At each check node we sum (modulo two) all these incoming values and store this (partial) sum in a memory cell which is associated to this check node. Further, we delete all involved edges. The result is shown in the right column entitled `right-to-left`. A black box indicates that the partial sum of this check node is currently one whereas a white box indicates a partial sum of zero. Note that there are three check nodes which have degree one. i.e., which received messages along all their edges *except one*. Since the modulo two sum of *all* messages into a particular check node is by definition of the code equal to zero, such a check node determines the value of the variable node it is connected to. Hence as a second step we propagate the partial sum of any check node which has currently degree one to its connected variable node. Again, a right-to-left message which is zero is indicated as a thin line, a message which is one is indicated as thick line and edges which do not carry any messages are drawn as dashed lines. The value of any variable node which receives a message along one of its edges is set to the value of this message and all involved edges are again deleted. The same procedure can now be repeated, i.e., we send the value of known variables along any of their remaining edges, accumulate partial sums at check nodes, delete all involved edges, send messages back from degree one check nodes, set the value of any so far unknown variable node which receives a message equal to the value of this message and again delete all involved edges. We will call one such round an *iteration* of the decoding algorithm.

If this procedure does not terminate prematurely then the value of all variable nodes will be determined. As can be seen from Fig. 5.10, for our example, the decoder recovers successfully from seven erasures after three iterations. But in general the decoder might fail in several ways: It might happen that for some iteration there is no check node of degree one so that no right-to-left message can be sent or that the right-to-left messages do not determine any so far unknown variable node so that no left-to-right message can be sent. In this case the decoding will remain incomplete. It is also easy to see that this decoder is suboptimal, i.e., that it will sometimes fail when a ML decoder will succeed, see Exercise 5.2.

It is important to notice that each edge is used at most once. Since the number of edges is proportional to (in our case three times) the number of variable nodes it follows that the decoding complexity is linear in the length of the code.

For the analysis of this decoding algorithm we will now reformulate the preceding algorithm as a *message passing* algorithm, i.e., an algorithm in which messages are passed along the edges of the graph.

Given a variable node $\mathtt{v}$ (a check node $\mathtt{c}$) let $\mathrm{E}(\mathtt{v})$ ($\mathrm{E}(\mathtt{c})$) be the set of its edges. Messages are from the set $\{0, 1, ?\}$ with a "?" indicating an *erasure*, i.e.,

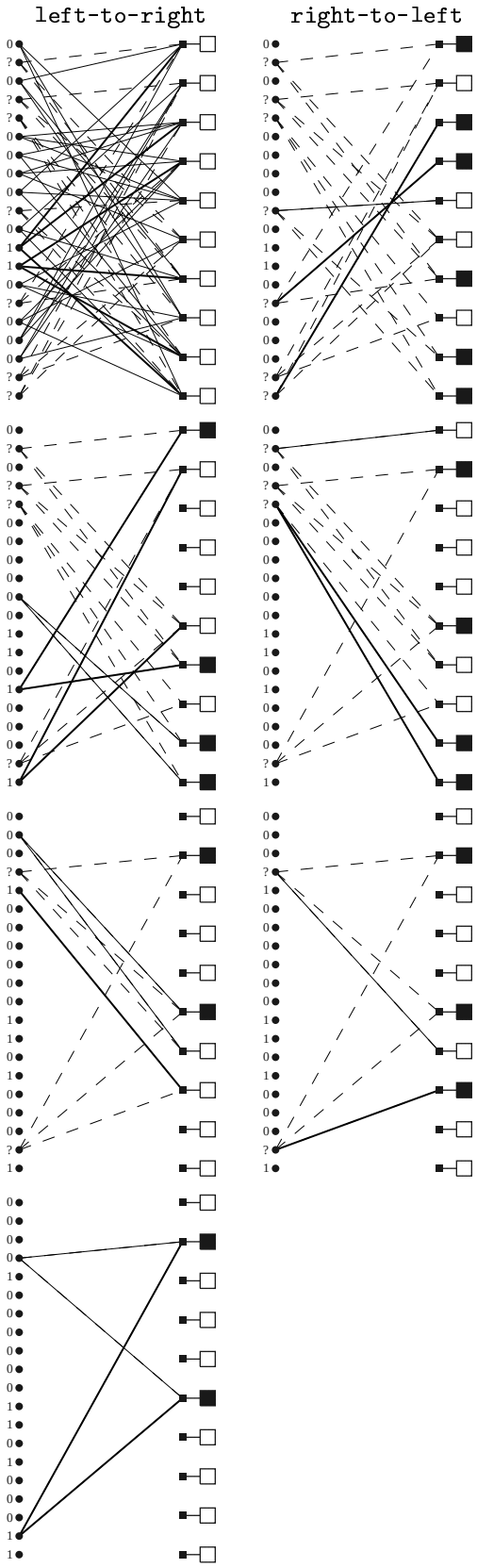


Figure 5.10: Iterative decoding for the erasure channel. After three iterations the decoder successfully recovers from seven erasures.

indicating that the corresponding bit has not been determined yet (along the given edge). At a variable node v the outgoing message along an edge e, $e \in E(v)$, is the erasure message if the received message associated to this node is an erasure and if incoming messages along all edges in $E(v) \setminus \{e\}$ are erasure messages, otherwise the outgoing message is equal to the value of this variable node. At a check node c the outgoing message along an edge e, $e \in E(c)$, is the erasure message if at least one of the incoming messages along all edges in $E(c) \setminus \{e\}$ is the erasure message, and it is the XOR of the incoming messages along all edges in $E(c) \setminus \{e\}$ otherwise. We say that a variable node is *known* if either its received value is known or if it has at least one incoming message which is not an erasure.

We claim that both description represent the same algorithm, i.e., that the set of known variable nodes at any iteration of the algorithm is the same. Let us apply this message passing algorithm to our example. This is shown in Fig. 5.11. We start with the *left-to-right* messages, conveying the values of the variable nodes to the check nodes. This is shown in the left-to-right column. In accordance with the previous example, zero messages are indicated as thin lines, one messages are shown as thick lines and erasure messages are indicated as dashed gray lines. The *right-to-left* messages are now determined from the previous left-to-right messages according to our rule. Messages are passed for several iterations until, hopefully, all variable nodes have been determined. By comparing Fig. 5.9 and Fig. 5.11 it is easy to check that at any iteration the set of *known* variables is the same for this example for both decoders. The proof of the claim that this is true in general is left as Exercise 5.3.

7. IRREGULAR LOW-DENSITY PARITY CHECK CODES

So far we have seen *regular* LDPC ensembles, i.e., LDPC codes all of whose variable (check) nodes have the same degree d_l (d_r). In order to achieve better performance it is necessary to regard *irregular* LDPC ensembles, we need to consider ensembles of codes whose nodes have various degrees. Therefore we define an *irregular* LDPC code as a LDPC code for which the degrees of each set of nodes are chosen according to some distribution. E.g., an irregular LDPC code might have a graphical representation in which half the variable nodes have degree 3 and half have degree 4, while half the constraint nodes have degree 6 and half have degree 8. Although the specification of the node degrees could be done in various ways the following notation leads to particularly elegant statements of many of the most fundamental results. In general, we call $\gamma(x)$ a *degree sequence* if $\gamma(x)$ is a real valued polynomial with non-negative coefficients and $\gamma(1) = 1$. Let d_l and d_r denote the maximum variable node and check node degrees, respectively, and let $\lambda(x) := \sum_{i=1}^{d_l} \lambda_i x^{i-1}$ and $\rho(x) := \sum_{i=1}^{d_r} \rho_i x^{i-1}$ be two degree sequences. More precisely, let the coefficients, λ_i (ρ_i) represent the fraction of *edges* emanating from variable (check) nodes of degree i. Then clearly this *degree sequence pair* (λ, ρ) completely specifies the distribution of the node degrees. The alert reader

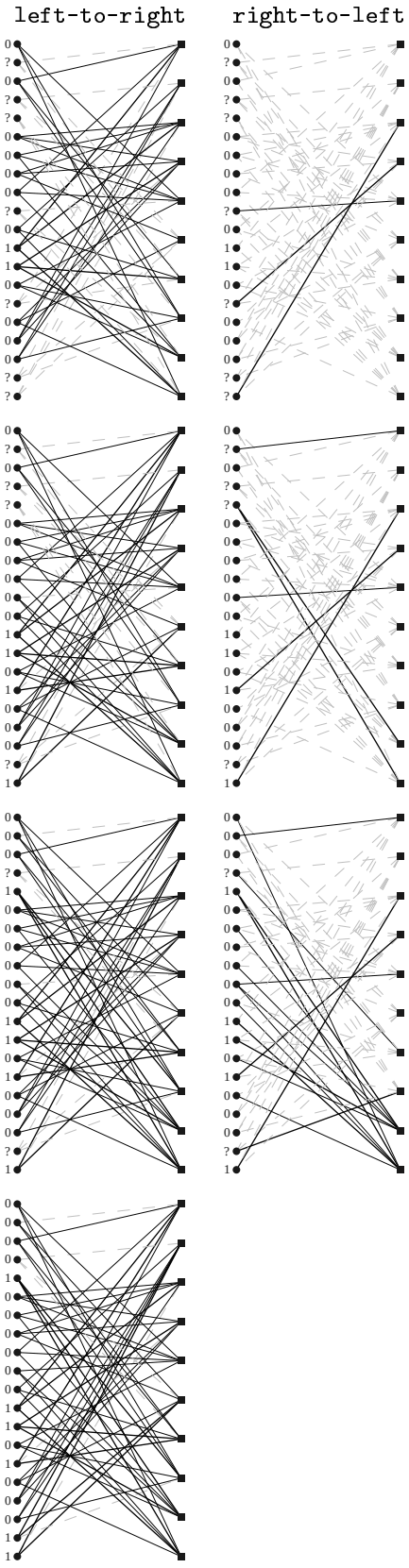


Figure 5.11: Message passing decoding for the erasure channel. After three iterations the decoder successfully recovers from seven erasures.

may have noticed several curious points about this notation. First, we do not specify the fraction of *nodes* of various degrees but rather the fraction of *edges* that emanate from nodes of various degrees. Clearly, it is easy to convert back and forth between this *edge perspective* and a *node perspective*. E.g., assume that half the variable nodes have degree three and half have degree four and that there is a total of n nodes. Since every degree three node has three edges emanating from it, whereas every degree four nodes has four edges emanating to it we see that there are in total $1/2 \cdot 3n$ edges which emanate from degree three nodes and that there are in total $1/2 \cdot 4n$ edges which emanate from degree four nodes. Therefore $\lambda_3 = \frac{1/2 \cdot 3}{1/2 \cdot 3 + 1/2 \cdot 4} = 3/7$ and $\lambda_4 = \frac{1/2 \cdot 4}{1/2 \cdot 3 + 1/2 \cdot 4} = 4/7$ so that in this case $\lambda(x) = 3/7x^2 + 4/7x^3$. Second, the fraction of edges which emanate from a degree i node is the coefficient of x^{i-1} rather than x^i as one might expect at first. The ultimate justification for this choice comes from the fact that, as we will see later, simple quantities like $\lambda'(0)$ or $\rho'(1)$ take on an operational meaning.

8. ANALYSIS OF DECODING ALGORITHM

Let x_0 denote the erasure probability of the channel and let x_ℓ denote the probability that a randomly chosen edge carries a left-to-right erasure message in the ℓ-th iteration. Given $x_{\ell-1}$ we would like to determine x_ℓ. Consider first a check node of degree d and the message emanating from this check node along a particular edge. This message will be an erasure message iff at least one incoming message along the other edges is an erasure message. This happens with probability $1 - (1 - x_{\ell-1})^{d-1}$. The probability that a randomly chosen edge is connected to a check node of degree d is given by ρ_d. Therefore the probability that a randomly chosen edge carries a right-to-left erasure message in the ℓ-th iteration is given by $1 - \rho(1 - x_{\ell-1})$.[1] To complete one iteration cycle consider now a variable node of degree d and a message which is emitted from it in the ℓ-th round along a particular edge. This message will be an erasure message if the incoming messages along *all* other edges as well as the received message are erasure messages. This happens with probability $x_0(1 - \rho(1 - x_{\ell-1}))^{d-1}$. Averaging over all variable node degrees we see that the probability of a right-to-left erasure message in the ℓ-th iteration along a randomly chosen edge is given by [**?**]

$$x_\ell = x_0 \lambda(1 - \rho(1 - x_{\ell-1})). \tag{5.1}$$

We will assume that we are only interested in coding systems for which the remaining erasure probability converges to zero (as the length of the codes tends to infinity). Therefore, for a fixed degree sequence pair (λ, ρ) we will be interested in how large we can choose the initial erasure probability x_0, i.e., how bad the channel can be, such that the expected fraction of erasures x_ℓ still converges to zero if we allow more and more iterations.

[1] Now we finally see the reason for specifying the various node degrees in this manner!

Note that for $x,y \in [0,1]$, the function $f(x,y) := y\lambda(1-\rho(1-x))$ is an increasing function in both of its variables. Assume that for some fixed $\epsilon \in [0,1]$, $x_\ell(\epsilon) \stackrel{\ell\to\infty}{\to} 0$. By finite induction we then see that for any $\epsilon' \in [0,\epsilon]$, $x_\ell(\epsilon') \stackrel{\ell\to\infty}{\to} 0$. This is true since by assumption $x_0(\epsilon') = \epsilon' \leq \epsilon = x_0(\epsilon)$, anchoring the finite induction, and since

$$x_\ell(\epsilon') = f(x_{\ell-1}(\epsilon'),\epsilon') \leq f(x_{\ell-1}(\epsilon'),\epsilon) \leq f(x_{\ell-1}(\epsilon),\epsilon) = x_\ell(\epsilon),$$

where in the one before last step we have used the induction hypothesis that $x_{\ell-1}(\epsilon') \leq x_{\ell-1}(\epsilon)$. Therefore it is meaningful to define the value x_0^* as the supremum of all values of x_0 such that x_ℓ converges to zero when ℓ tends to infinity. There are many other equivalent characterizations of x_0^*. A nice graphical characterization of x_0^* is given as follows. Note that in order for $x_\ell(\epsilon)$ to converge to zero we need that x_ℓ is strictly decreasing, i.e., $x_\ell - x_{\ell-1} = x_0\lambda(1-\rho(1-x_{\ell-1})) - x_{\ell-1} < 0$. We can therefore look for the supremum of all x_0 such that $x_0\lambda(1-\rho(1-x)) - x$ is strictly negative for $0 \leq x \leq x_0$. This is shown graphically in Fig. 5.12 for the ensemble of $(3,6)$-regular codes and the values $x_0 = 0, 4, 0.42944, 0.45$. We see that the supremum of all x_0 such that this plot is strictly negative over the whole range of interest is taken on around 0.42944. Note that for $x_0 \sim 0.42944$ there is one *critical* value for which the expected decrease in the erasure fraction is zero.

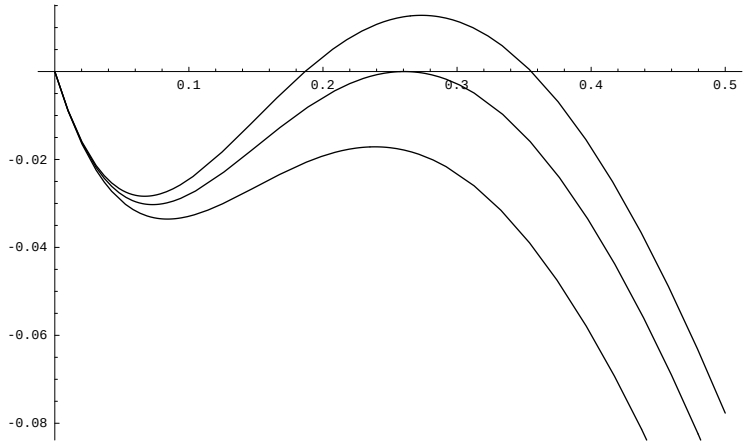


Figure 5.12: Plots of the function $x_0\lambda(1-\rho(1-x))-x$ for $x_0 = 0, 4, 0.42944, 0.45$. We see that the supremum of all x_0 such that this plot is strictly negative over the whole range of interest is taken on around 0.42944.

8.1 ANALYTIC DETERMINATION OF THE THRESHOLD

Although one could use the above graphical method to determine the threshold to any degree of accuracy it is nevertheless pleasing (and not too hard) to derive analytic expressions for these thresholds. Solving equation (**??**) for x_0 we get

$$x_0(x) = \frac{x}{\lambda(1-\rho(1-x))}. \tag{5.2}$$

In words, for a given positive real number x there is a unique value of x_0 such that x fulfils the fixed point equation (**??**). By the remarks above, if $x \leq x_0$ then the threshold is upper bounded by x_0. This fact is used in the following lemma to determine the threshold of regular codes for the BEC.

Lemma 9. [Thresholds for the BEC and Regular Codes] Assume we are given a $(\mathrm{d_l}, \mathrm{d_r})$-regular LDPC code. Let σ be the unique positive real root of the polynomial $p(x) = ((\mathrm{d_l}-1)(\mathrm{d_r}-1)-1)x^{\mathrm{d_r}-2} - \sum_{i=0}^{\mathrm{d_r}-3} x^i$. Then the threshold $x_0^*(\mathrm{d_l}, \mathrm{d_r})$ is equal to $x_0^*(\mathrm{d_l}, \mathrm{d_r}) = \frac{1-\sigma}{(1-\sigma^{\mathrm{d_r}-1})^{\mathrm{d_l}-1}}$.

Proof. By the remarks above, the threshold is given by $\min\{g(x) : g(x) \geq x\}$, where $g(x) := \frac{x}{\lambda(1-\rho(1-x))} = \frac{x}{(1-(1-x)^{\mathrm{d_r}-1})^{\mathrm{d_l}-1}}$. First note that $g(x) \geq x$ with equality only at $x = 1$. It follows that the threshold is given by the minimum of $g(x)$ over the range $x \in [0,1]$. In the sequel it will be slightly more convenient to consider $g(1-x) = \frac{1-x}{(1-x^{\mathrm{d_r}-1})^{\mathrm{d_l}-1}}$. Taking the derivative of $g(1-x)$ with respect to x and setting the result to zero we get the polynomial equation

$$((\mathrm{d_r}-1)(\mathrm{d_l}-1)-1)x^{\mathrm{d_r}-1} - (\mathrm{d_r}-1)(\mathrm{d_l}-1)x^{\mathrm{d_r}-2} + 1 = 0.$$

By Descartes' rule of signs[2] this equation has at most two positive real roots. Clearly, one such root is at $x = 1$. Hence, dividing by $x-1$ we get the simplified polynomial equation

$$((\mathrm{d_l}-1)(\mathrm{d_r}-1)-1)x^{\mathrm{d_r}-2} - \sum_{i=0}^{\mathrm{d_r}-3} x^i = 0,$$

which by the above remark has at most a single l positive root. It is easy to see that $g(x)$ has a pole at $x = 0$ and is equal to 1 at $x = 1$. Hence, $g(x)$ does not take on its minimum at the boundary. It follows that it must take on its minimum in the interior, hence, at least one and therefore exactly one solution to the above polynomial equation must exist. □

[2] Descartes' rule of signs bounds the number of positive real roots of a polynomial $f(x)$ in one variable. If

$$f(x) = \sum_{j=1}^{r} c_j x^{m_j},$$

with $0 \leq m_1 < m_2 < \ldots < m_r$ and with all coefficients $c_j \neq 0$, then the number of positive real zeros of f is upper bounded by the number of sign changes $N^+(f)$ between consecutive coefficients c_j when taken in order of increasing j, see [**?**], [**?**, Chapter 6].

In the following examples the threshold for some standard regular codes are determined.

Example 19. [$(3,6)$ code] Let σ be given by

$$\sigma = \frac{1}{36} + \frac{\sqrt{\frac{25}{324} - a + b}}{2} + \frac{\sqrt{\frac{25}{162} + a - b + \frac{685}{2916\sqrt{\frac{25}{324} - a + b}}}}{2},$$

where $a = \frac{22}{27} 5^{\frac{2}{3}} \left(\frac{2}{-85+3\sqrt{24465}} \right)^{\frac{1}{3}}$ and $b = \frac{1}{27} \left(\frac{5}{2} \left(-85 + 3\sqrt{24465} \right) \right)^{\frac{1}{3}}$. Then $x_0^*(3,6) := \frac{1-\sigma}{(1-\sigma^5)^2} \sim 0.42944$.

Example 20. [$(3,5)$ code] $x_0^*(3,5) = \dfrac{1 + \frac{-1 - \left(694 - 42\sqrt{267}\right)^{\frac{1}{3}} - \left(694 + 42\sqrt{267}\right)^{\frac{1}{3}}}{21}}{\left(1 - \frac{\left(1 + \left(694 - 42\sqrt{267}\right)^{\frac{1}{3}} + \left(694 + 42\sqrt{267}\right)^{\frac{1}{3}} \right)^4}{194481} \right)^2} \sim 0.540605.$

Example 21. [$(4,6)$ code] Let σ be given by

$$\sigma = \frac{1}{56} + \frac{1}{2}\sqrt{\frac{115}{2352} - a + b} + \frac{1}{2}\sqrt{\frac{115}{1176} + a - b + \frac{1625}{10976\frac{115}{2352} - a + b}},$$

where $a = \frac{175^{2/3}}{21(-65+3\sqrt{22305})^{1/3}}$ and $b = \frac{1}{42}(5(-65+3\sqrt{22305}))^{1/3}$. Then $x_0^*(4,6) := \frac{1-\sigma}{(1-\sigma^5)^3} \sim 0.506741$.

Example 22. [$(3,4)$ code] $x_0^*(3,4) = \frac{3125}{3672+252\sqrt{21}} \sim 0.647426$.

8.2 THE STABILITY CONDITION

Consider again the recursion $x_\ell = x_0\lambda(1-\rho(1-x_{\ell-1}))$ which describes the evolution of the expected fraction of erasure messages on an infinite tree. Although in the previous section we saw that, at least for regular codes, it is quite easy to give an analytic expression for the threshold x_0^* of such codes it is nevertheless instructive to derive the following upper bound on x_0^*. This upper bound is derived by looking at the behavior of the above recursion for small values of x_ℓ. We will see later that the same principle can be used to derive upper bounds on the threshold for general memoryless channels in which case there is currently no known method for their analytic determination.

Let $h(x) := x_0\lambda(1-\rho(1-x))$. Expanding $h(x)$ in powers of x we see that $h(x) = x_0\lambda'(0)\rho'(1)x + O(x^2)$. Therefore, to first order in x, the fraction of erasure messages will evolve from x to $x_0\lambda'(0)\rho'(1)x$. Clearly, if we want the fraction of erasure messages to tend to zero then we need $\lambda'(0)\rho'(1) < 1/x_0$. From this we can deduce the bound $x_0^* < \frac{1}{\lambda'(0)\rho'(1)}$. Vice versa, if $\lambda'(0)\rho'(1) < 1/x_0$ then there

exists an $x > 0$ such that the values of the recursion tend to zero if the recursion is initialized with a value which does not exceed x. The condition $\lambda'(0)\rho'(1) < 1/x_0$, first discussed in [1], can be seen as a *stability condition* of the fixed point at $x = 0$.

9. GENERAL CHANNELS

The basic principle of iterative decoding is the same for general types of channels. The only change required is to adapt the message alphabet and the variable node and check node maps. By choosing these parameters appropriately one can explore a large section of the complexity versus performance plane. Fig. 5.13 shows the achievable performance for the BIAWGNC.

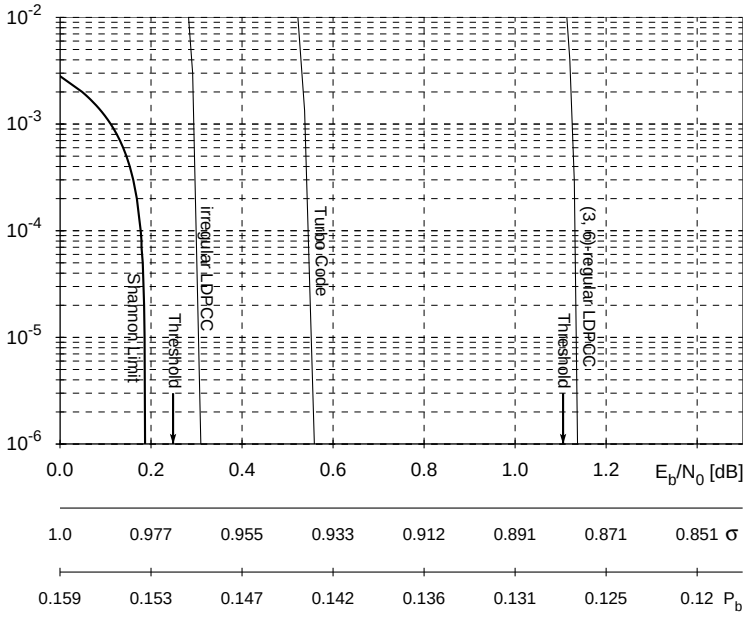


Figure 5.13: Comparison of an $(3,6)$-regular LDPC code, turbo code, and optimized irregular LDPC code. All codes are of length 10^6 and of rate one-half. The bit error rate for the BIAWGNC is shown as a function of E_b/N_0 (in dB), the standard deviation σ, as well as the raw input bit error probability P_b.

EXERCISES

5.1. In this example we will explore some of the basic properties of binary linear block codes.

1. Convince yourself that $\mathrm{GF}(2)^n$ is a vector space.

2. A binary linear block code is a subspace of $\mathrm{GF}(2)^n$ for some n and therefore has a dimension k, $0 \leq k \leq n$. We can therefore represent such a code $\mathcal{C}$ as

$$\mathcal{C} := \{x \in \mathrm{GF}(2)^n : x = uG; u \in \mathrm{GF}(2)^k\},$$

   where $G \in \mathrm{GF}(2)^{k \times n}$ is called the *generator matrix*. Define the set of words $\mathcal{C}^\perp$ as

$$\mathcal{C}^\perp := \{y \in \mathrm{GF}(2)^n : Gy^T = 0^T\}.$$

   Show the following:

   (a) $\mathcal{C}^\perp$ is a linear subspace of $\mathrm{GF}(2)^n$.

   (b) The dimension of $\mathcal{C}^\perp$ is $n-k$.

   (c) From the above two observations conclude that $\mathcal{C}^\perp$ has a representation of the form

$$\mathcal{C}^\perp := \{x \in \mathrm{GF}(2)^n : x = uH; u \in \mathrm{GF}(2)^{n-k}\}.$$

   (d) Show that $x \in \mathcal{C}$ if and only if $Hx^T = 0^T$. H is called the *parity check matrix*.

3. As we have seen in class, the generator matrix for the n-repetition code is equal to

$$G = \underbrace{(1, \ldots, 1)}_{n\times}.$$

   What is the corresponding parity check matrix?

4. As a second example, consider the so-called *single-parity check code* of length n which has a generator matrix equal to

$$G = \begin{pmatrix} 1 & 0 & 0 & \ldots & 0 & 0 & 1 \\ 0 & 1 & 0 & \ldots & 0 & 0 & 1 \\ \vdots & \ldots & \ldots & \ldots & \ldots & \ldots & 1 \\ 0 & 0 & 0 & \ldots & 0 & 1 & 1 \end{pmatrix}$$

   What is it's associated parity-check matrix? Any comments?

5. The code is said to be in *systematic form* if the first $k \times k$ submatrix of G is the diagonal identity matrix. Assuming that G is in systematic form, can you easily find a corresponding H?

5.2. Show that the message passing decoder for the BEC is suboptimal by finding a simple graph and a particular codeword such that the ML decoder will succeed but such that the iterative algorithm will fail. What is the smallest example you can find?

5.3. Show that the two iterative decoders for the BEC are indeed equivalent, i.e., that for any iteration the set of known variable nodes is the same.

6

SOLUTIONS OF THE EXERCISES - 1

Exercise 1.1
Recall that the random variables $Z_1, \cdots, Z_n$ are jointly Gaussian when they form a (jointly) Gaussian random vector $(Z_1, \cdots, Z_n)$. Here, the covariance matrix of the zero-mean Gaussian random vector $(Z_1, \cdots, Z_n)$ is $\sigma^2 I_n$ (the matrix I_n denoting the $n \times n$ identity matrix). Consider P the (constant) matrix representing the transformation of the canonical basis into the orthonormal basis $(\phi_1, \cdots, \phi_n)$. Both basis are orthonormal, therefore P is orthogonal, i.e., $PP^T = I_n$. Since $W = PZ$, the linear mapping induces that, like Z, W is a zero-mean Gaussian random vector. Moreover, it has covariance $\mathbb{E}W^T W = \mathbb{E}(PZ)^T (PZ) = \mathbb{E}Z^T P^T Z = \mathbb{E}Z^T Z = \sigma^2 I_n$. Therefore, W has same distribution as Z.

Exercise 4.2
In the Gaussian case with uniform priors, the decision regions are the Voronoi regions:
Consider m points $a_i \in \mathbb{R}^n$ with uniform priors $p_i = \frac{1}{m}$. Under hypothesis i, the observation Y is $Y = a_i + Z$, where $Z = (Z_1, \cdots, Z_n)$ is a jointly Gaussian vector of independent zero mean random variables each of variance σ^2. With

$$f_{Y|H}(y|i) = \frac{1}{(2\pi\sigma^2)^{n/2}} e^{-\frac{\|y-a_i\|^2}{2\sigma^2}},$$

the MAP decision rule is

$$\begin{aligned} \hat{H}(y) &= \text{argmax}_i \Big(f_{Y|H}(y|i) p_i \Big) \\ &= \text{argmax}_i \Big(-\frac{\|y-a_i\|^2}{2\sigma^2} + \ln(p_i) \Big) \\ &= \text{argmin}_i \Big(\frac{\|y-a_i\|^2}{2\sigma^2} - \ln(\frac{1}{m}) \Big) \\ &= \text{argmin}_i \|y-a_i\|, \end{aligned} \tag{6.1}$$

so that the decision regions are equal to the Voronoi regions.

Convexity of the decision regions in the Gaussian case:
An element x is in the Voronoi region associated to the hypothesis i iff, for all $j \neq i$, $\|x-a_i\|^2 \leq \|x-a_j\|^2$, or, iff, for all $j \neq i$, $< x, a_j - a_i > \leq \frac{\|a_j\|^2 - \|a_i\|^2}{2}$.
Consider x_1, x_2 two elements of the Voronoi region associated to the hypothesis i. Taking $\alpha \in [0,1]$, we can write,

$$\begin{aligned} < \alpha x_1 + (1-\alpha)x_2, a_j - a_i > &= \alpha < x_1, a_j - a_i > + (1-\alpha) < x_2, a_j - a_i > \\ &\leq \alpha \frac{\|a_j\|^2 - \|a_i\|^2}{2} + (1-\alpha)\frac{\|a_j\|^2 - \|a_i\|^2}{2} \\ &= \frac{\|a_j\|^2 - \|a_i\|^2}{2}, \end{aligned} \tag{6.2}$$

for all $j \neq i$. I.e., the point $\alpha x_1 + (1-\alpha)x_2$ is in the Voronoi region associated to a_i.

Exercise 1.3
(i) *Exact probabilty or error for MAP*
The 8 points of the modulation scheme are equivalent and we therefore have

$$\Pr\{error\} = \Pr\{e|H=4\} = \Pr\{z \in B\} = 2\Pr\{z \in A\},$$

as shown in Fig. 6.1. Changing to polar coordinates as indicated in Fig. 6.2

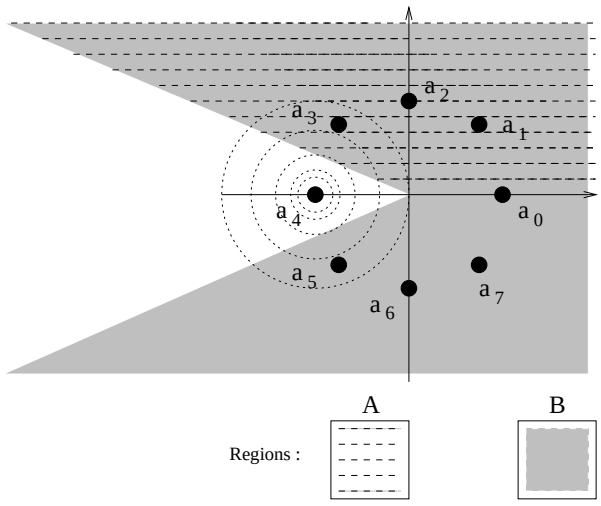


Figure 6.1: Integration region for error probability

we get,

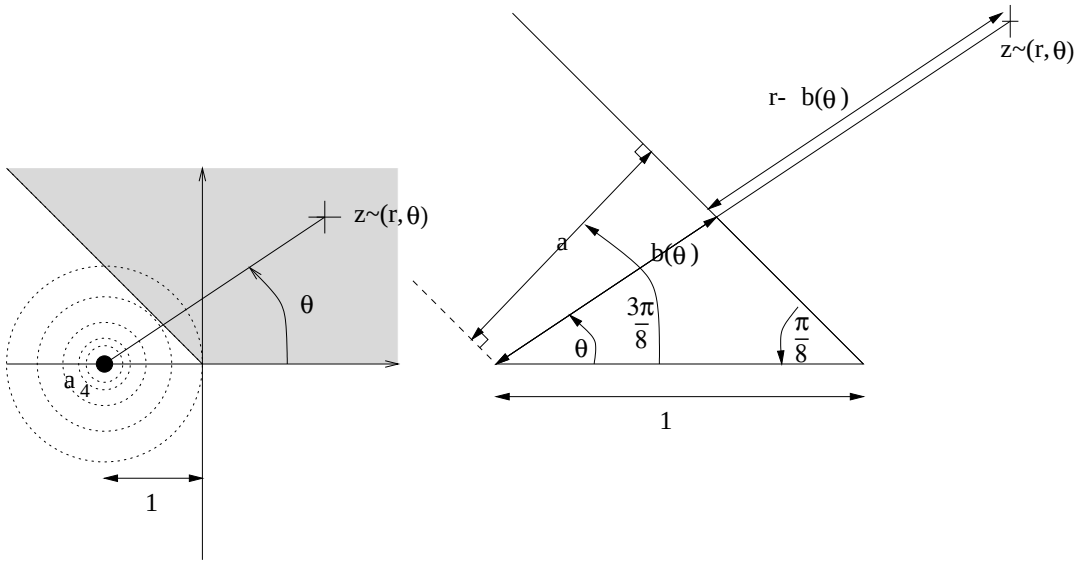


Figure 6.2: Polar coordinates system for integration

$$\Pr\{z \in A\} = \int_{\theta=0}^{\frac{7\pi}{8}} \int_{r=b(\theta)}^{+\infty} \frac{1}{2\pi\sigma^2} \exp(-\frac{r^2}{2\sigma^2}) r \mathrm{d}\theta \mathrm{d}r. \tag{6.3}$$

Computation of $b(\theta)$: From Fig. 6.2 we see that $a = \sin(\frac{\pi}{8})$. Therefore,

$$b(\theta)\cos(\frac{3\pi}{8} - \theta) = \sin(\frac{\pi}{8}) \Rightarrow b(\theta) = \frac{\sin(\frac{\pi}{8})}{\sin(\theta + \frac{\pi}{8})}.$$

Computation of $Pr\{z \in A\}$:

$$\begin{aligned} \Pr\{z \in A\} &= \int_{\theta=0}^{\frac{7\pi}{8}} \int_{r=b(\theta)}^{+\infty} \frac{1}{2\pi\sigma^2} \exp(-\frac{r^2}{2\sigma^2}) r \mathrm{d}\theta \mathrm{d}r \\ &= \int_{\theta=0}^{\frac{7\pi}{8}} \Big[-\frac{1}{2\pi} \exp(-\frac{r^2}{2\sigma^2}) \Big]_{b(\theta)}^{+\infty} \mathrm{d}\theta \\ &= \int_{\theta=0}^{\frac{7\pi}{8}} \frac{1}{2\pi} \exp(-\frac{b(\theta)^2}{2\sigma^2}) \mathrm{d}\theta. \end{aligned}$$

Conclusion: The exact probability of error is given by

$$\Pr\{error\} = 2\Pr\{z \in A\} = \frac{1}{\pi} \int_{\theta=0}^{\frac{7\pi}{8}} \exp\Big(-\frac{\sin^2(\frac{\pi}{8})}{2\sigma^2 \sin^2(\theta + \frac{\pi}{8})} \Big) \mathrm{d}\theta. \tag{6.4}$$

(ii) *Union bound on probabilty of error for MAP*
Consider the situation depicted in Fig. 6.3. There are two intersecting regions C (right to the axis y=-x) and D (right to the axis y=x). We have,

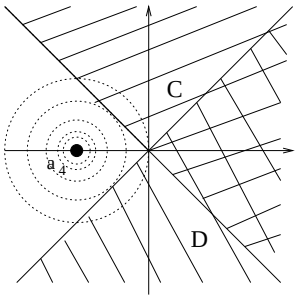


Figure 6.3: Integration region for error probability using the union bound

$$\begin{aligned}
\Pr\{error\} &= \Pr\{e|H=4\} \\
&= \Pr\{z \in C \cup D\} \\
&\leq \Pr\{z \in C\} + \Pr\{z \in D\} \\
&= Q(\frac{\sin(\frac{\pi}{8})}{\sigma}) + Q(\frac{\sin(\frac{\pi}{8})}{\sigma}) \\
&= 2Q(\frac{\sin(\frac{\pi}{8})}{\sigma}).
\end{aligned}$$

Exercise 1.4

Formula for the MAP decision rule:

Denoting A the random variable for the transmitted points, the hypothesis $H = i$, $i \in \{0,1\}$, leads to transmit $A = a_{i0}$ or $A = a_{i1}$ with equal probability. Therefore, the MAP decision rule is

$$\begin{aligned}
\hat{H}(y) &:= \text{argmax}_i p_{H|Y}(i|y) \\
&= \text{argmax}_i \Big(p_{A|Y}(a_{i0}|y) + p_{A|Y}(a_{i1}|y) \Big) \\
&= \text{argmax}_i \Big(f_{Y|A}(y|a_{i0}) + f_{Y|A}(y|a_{i1}) \Big)
\end{aligned}$$

where the last equality occurs from the uniform priors for A. (For $(k,l) \in \{0,1\}^2$, $\Pr\{A = a_{kl}\} = 1/4$ since the priors for H are $1/2$ and the two corresponding possible transmitted points are then equally likely.). More explicitly, one can write,

$$\begin{aligned}
\hat{H}(y) &= \text{argmax}_{i \in \{0,1\}} \Big(e^{-\frac{\|y-a_{i,0}\|^2}{2\sigma^2}} + e^{-\frac{\|y-a_{i,1}\|^2}{2\sigma^2}} \Big) \\
&= \begin{cases} 0 & \text{if } \mathcal{W}(y) > 1, \\ 1 & \text{if } \mathcal{W}(y) < 1, \end{cases}
\end{aligned}$$

where $\mathcal{W}(y)$ indicates the quotient $\mathcal{W}(y) := \frac{e^{-\frac{\|y-a_{0,0}\|^2}{2\sigma^2}}+e^{-\frac{\|y-a_{0,1}\|^2}{2\sigma^2}}}{e^{-\frac{\|y-a_{1,0}\|^2}{2\sigma^2}}+e^{-\frac{\|y-a_{1,1}\|^2}{2\sigma^2}}}$.

Geometric interpretation:
Consider the coordinates system (O,u,v) for the 2-dimensional plane containing the received point y and the possible transmitted points $\{a_{kl}\}_{(k,l)\in\{0,1\}^2}$. The coordinates couple for y is (y_u,y_v). We have,

$$\begin{aligned}\mathcal{W}(y) &= \mathcal{W}(y) = \frac{e^{-\frac{\|y-a_{0,0}\|^2}{2\sigma^2}}+e^{-\frac{\|y-a_{0,1}\|^2}{2\sigma^2}}}{e^{-\frac{\|y-a_{1,0}\|^2}{2\sigma^2}}+e^{-\frac{\|y-a_{1,1}\|^2}{2\sigma^2}}}\\ &= \frac{e^{-\frac{y_u^2+y_v^2+2}{2\sigma^2}}\left(e^{\frac{y_u+y_v}{\sigma^2}}+e^{-\frac{x_u+y_v}{\sigma^2}}\right)}{e^{-\frac{x_u^2+y_v^2+2}{2\sigma^2}}\left(e^{\frac{-x_u+y_v}{\sigma^2}}+e^{\frac{x_u-y_u}{\sigma^2}}\right)}\\ &= \frac{1+e^{-\frac{2y_u}{\sigma^2}}e^{-\frac{2y_v}{\sigma^2}}}{e^{-\frac{2y_u}{\sigma^2}}+e^{-\frac{2y_v}{\sigma^2}}}.\end{aligned}$$

The $\mathcal{W}(y)$ is here recognized, for $y_u \geq 0$ and $y_v \geq 0$, as being on the form

$$\mathcal{W}(y) = \frac{1}{\tanh\left(\operatorname{atanh}(e^{-\frac{2y_u}{\sigma^2}})+\operatorname{atanh}(e^{-\frac{2y_v}{\sigma^2}})\right)} \geq 1.$$

Indeed, in that case, $e^{-\frac{2y_u}{\sigma^2}} \leq 1$ and $e^{-\frac{2y_v}{\sigma^2}} leq1$ can be viewed as values of tangent hyperbolicus functions. More generally, depending on the sign of y_u and y_v, the quotient $\mathcal{W}(y)$ can always be written as $1/\tanh\left(\operatorname{atanh}(a)+\operatorname{atanh}(b)\right)$ or as $\tanh\left(\operatorname{atanh}(a)+\operatorname{atanh}(b)\right)$. Finally, it follows that,

$$\begin{aligned}\hat{H}(y) &= \operatorname{argmax}_{i\in\{0,1\}}\left(e^{-\frac{\|y-a_{i,0}\|^2}{2\sigma^2}}+e^{-\frac{\|y-a_{i,1}\|^2}{2\sigma^2}}\right)\\ &= \begin{cases}0 & \text{if } (y_u \geq 0 \text{ and } y_v \geq 0) \text{ or if } (y_u \leq 0 \text{ and } y_v \leq 0),\\ 1 & \text{if } (y_u \leq 0 \text{ and } y_v \geq 0) \text{ or if } (y_u \geq 0 \text{ and } y_v \leq 0).\end{cases}\end{aligned}$$

The decision regions are represented in Fig. 6.4.

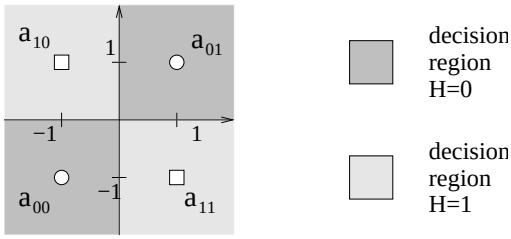


Figure 6.4: Hypothesis testing for H.

Exercise 1.5
Let $x \geq 0$. Then

$$\begin{aligned} Q(x) &= \int_x^\infty \frac{1}{\sqrt{2\pi}} e^{-\frac{z^2}{2}} dz \\ &= e^{-\frac{x^2}{2}} \int_x^\infty \frac{1}{\sqrt{2\pi}} e^{-\frac{(z^2-x^2)}{2}} dz \\ &\overset{z\geq x}{\leq} e^{-\frac{x^2}{2}} \int_x^\infty \frac{1}{\sqrt{2\pi}} e^{-\frac{(z-x)^2}{2}} dz \\ &\leq e^{-\frac{x^2}{2}} \int_{-\infty}^\infty \frac{1}{\sqrt{2\pi}} e^{-\frac{(z-x)^2}{2}} dz \\ &= e^{-\frac{x^2}{2}}. \end{aligned}$$

The inequality can also be obtained by considering the function $f(x) = \mathrm{e}^{-\frac{x^2}{2}} - Q(x)$. Note that $f(0) = \frac{1}{2} > 0$. Further, since $f'(x) = \mathrm{e}^{-\frac{x^2}{2}}(\frac{1}{\sqrt{2\pi}} - x)$ is negative for $x \geq 0$ it follows that $f(x)$ is a nonincreasing for $x \geq 0$. The claim now follows by observing that $\lim_{x\to\infty} f(x) = 0^+$.
Let $x \geq 0$. Then

$$\begin{aligned} xQ(x) &= \int_x^\infty \frac{x}{\sqrt{2\pi}} e^{-\frac{z^2}{2}} dz \\ &\overset{z\geq x}{\leq} \int_x^\infty \frac{z}{\sqrt{2\pi}} e^{-\frac{z^2}{2}} dz \\ &= \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}. \end{aligned}$$

Exercise 1.6
The noise component Z_k for $k = 1, 2$ is a Gaussian random variables since $Z(t)$ is a Gaussian random variable and $\psi_k(t)$ is deterministic. Its mean values is

$$\mathrm{E}[Z_k] = \int_{-\infty}^{+\infty} \mathrm{E}[Z(t)] = 0.$$

The covariances of the noise components are, for $i, j \in \{1,2\}$,

$$\begin{aligned}
\mathrm{E}[Z_i Z_j] &= \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} \mathrm{E}[Z(t)Z(t')]\psi_i(t)\psi_j(t')\mathrm{d}t\mathrm{d}t' \\
&= \frac{N_0}{2}\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} \delta(t-t')\psi_i(t)\psi_j(t')\mathrm{d}t\mathrm{d}t' \\
&= \frac{N_0}{2}\int_{-\infty}^{+\infty} \psi_i(t)\psi_j(t)\mathrm{d}t \\
&= \frac{N_0}{2}\delta_{i,j}.
\end{aligned}$$

Therefore the distribution of the random vector (Z_1, Z_2) is a zero-mean Gaussian with covariance matrix $\frac{N_0}{2} I_2$.

Exercise 1.7

The convolution operation is a linear operation performed on the input signal $X(t)$. Therefore the expected value of the integrale is the integrale of the expected value.

Y(t) is a zero-mean process:
We have $\mathrm{E}[X(t)] = 0$, then

$$\begin{aligned}
\mathrm{E}[Y(t)] &= \mathrm{E}\Big[\int_{-\infty}^{\infty} X(\tau)h(t-\tau)\mathrm{d}\tau\Big] \\
&= \int_{-\infty}^{\infty} \mathrm{E}[X(\tau)]h(t-\tau)\mathrm{d}\tau \\
&= 0.
\end{aligned}$$

Covariance function of Y(t):
In the expression

$$\begin{aligned}
K_Y(t,u) &= \mathrm{E}[Y(t)Y(u)] \\
&= \mathrm{E}\Big[\int_{-\infty}^{\infty} X(\tau)h(t-\tau)\mathrm{d}\tau \int_{-\infty}^{\infty} X(\tau')h(u-\tau')\mathrm{d}\tau'\Big] \\
&= \mathrm{E}\Big[\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} X(\tau)h(t-\tau)X(\tau')h(u-\tau')\mathrm{d}\tau\mathrm{d}\tau'\Big] \\
&= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \mathrm{E}[X(\tau)X(\tau')]h(t-\tau)h(u-\tau')\mathrm{d}\tau\mathrm{d}\tau' \\
&= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} K_X(\tau,\tau')h(t-\tau)h(u-\tau')\mathrm{d}\tau\mathrm{d}\tau',
\end{aligned}$$

which clearly *depends on* K_X *and* $h(t)$.
Assume now the process is wide sense stationary, i.e., $K_X(t,u) = K_X(t-u)$,

then, we make the change $\hat{\tau} = t - \tau$ and $\hat{\tau'} = u - \tau'$, so that,

$$\begin{aligned} K_Y(t,u) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} K_X(t-\hat{\tau}, u-\hat{\tau'})h(\hat{\tau})h(\hat{\tau'})\mathrm{d}\hat{\tau}\mathrm{d}\hat{\tau'} \\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} K_X(t-u+\hat{\tau'}-\hat{\tau})h(\hat{\tau})h(\hat{\tau'})\mathrm{d}\hat{\tau}\mathrm{d}\hat{\tau'}. \end{aligned}$$

The covariance $K_Y(t,u)$ is a function only of $t-u$, i.e, we can only consider the function $K_Y(\delta)$

Relation between power spectral density of $X(t)$ and $Y(t)$:
We have

$$\begin{aligned} \mathcal{S}_Y(f) &= \int_{-\infty}^{\infty} K_Y(\delta)\mathrm{e}^{-j2\pi f\delta}\mathrm{d}\delta \\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} K_X(\delta+\hat{\tau'}-\hat{\tau})h(\hat{\tau})h(\hat{\tau'})\mathrm{e}^{-j2\pi f\delta}\mathrm{d}\hat{\tau}\mathrm{d}\hat{\tau'}\mathrm{d}\delta \\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h(\hat{\tau})h(\hat{\tau'})\mathrm{e}^{-j2\pi f\hat{\tau}}\mathrm{e}^{-j2\pi f\hat{\tau'}}\int_{-\infty}^{\infty} K_X(\delta')\mathrm{e}^{-j2\pi f\delta'}\mathrm{d}\delta'\mathrm{d}\hat{\tau}\mathrm{d}\hat{\tau'}, \end{aligned}$$

using $\delta' = \delta + \hat{\tau'} - \hat{\tau}$, i.e,

$$\mathcal{S}_Y(f) = |H(f)|^2\mathcal{S}_X(f).$$

The output signal is the product of the power density spectrum of the input multiplied by the magnitude squared of the frequency response.

Exercise 1.8
We consider jointly wide-sense stationary processes with cross-correlation function $\mathcal{R}_{XU}(\tau)$. We have,

$$\begin{aligned} \mathcal{R}_{XU}(\tau) &= \mathbb{E}[X(t)U^*(t-\tau)] \\ &= \mathbb{E}[(X^*(t)U(t-\tau))^*] \\ &= \Big(\mathbb{E}[X^*(t)U(t-\tau)]\Big)^* \\ &= \Big(\mathbb{E}[U(t-\tau)X^*(t)]\Big)^* \\ &\overset{(a)}{=} \Big(\mathbb{E}[U(\tilde{t})X^*(\tilde{t}+\tau)]\Big)^* = \mathcal{R}^*_{UX}(-\tau), \end{aligned}$$

using in (a) the stationarity of the joint process $U(\tilde{t})X^*(\tilde{t}+\tau)$.

Exercise 1.9
The process $Y(t)$ has is a zero-mean Gaussian process since it is simply a linear combination of the zero-mean Gaussian process $Z(t)$. Moreover, for all t and τ, we have,

$$\begin{aligned} \mathbb{E}[Z(t)Z(t-\tau)] &= \mathbb{E}[\mathrm{e}^{2\pi f_0 t}Z(t)\mathrm{e}^{2\pi f_0(t-\tau)}Z(t-\tau)] \\ &= \mathrm{e}^{2\pi f_0(2t-\tau)}\mathbb{E}[Z(t)Z(t-\tau)] = 0, \end{aligned}$$

since $\mathbb{E}[Z(t)Z(t-\tau)] = 0$. I.e., $Y(t)$ is circularly symmetric.
Such a process is stationnary if and only it is wide-sense stationary. Since $\mathbb{E}[Y(t)] = e^{2\pi f_0 t}\mathbb{E}[Z(t)]$ and $\mathbb{E}[Y(t+\tau)] = e^{2\pi f_0(t+\tau)}\mathbb{E}[Z(t+\tau)] = e^{2\pi f_0(t+\tau)}\mathbb{E}[Z(t)] = 0$ are independent of t, it is also stationary.

Exercise 1.10

(a) For $\mathcal{S}_{R_1}$ (respectively $\mathcal{S}_{R_2}$, $\mathcal{S}_{R_3}$) the average energy of the constellation is denoted $\overline{E}^{R_1}(d)$ (respectively $\overline{E}^{R_2}(d)$,$\overline{E}^{R_3}(d)$). It follows

$$\begin{aligned}
\overline{E}^{R_1}(d) &= 2d^2, \\
\overline{E}^{R_2}(d) &= \frac{1}{16}[4(2d^2) + 8(10d^2) + 4(18d^2)] = 10d^2, \\
\overline{E}^{R_3}(d) &= \frac{1}{32}[16(10d^2) + 8(25d^2) + 8(34d^2)] = 20d^2,
\end{aligned}$$

assuming that all points are equally likely.

We investigate now how the number of signal points and the average energy scale with R under the continuous approximation "$\frac{\pi}{2d^2} r\delta r$ represents the number of grid points which have norm between r and $r+\delta r$

(b) Denote $N(R)$ the number of grid points within radius R and $\overline{E}^{R}_{cont}(d)$ the average energy of all grid points within radius R, assuming that all points are equally likely. We get,

$$\begin{aligned}
N(R) &= \int_{r=0}^{R} \frac{\pi}{2d^2} r\mathrm{d}r, \\
\overline{E}^{R}_{cont}(d) &= \frac{1}{N(R)} \int_{r=0}^{R} r^2 \cdot \frac{\pi}{2d^2} r\mathrm{d}r = \frac{\frac{\pi R^4}{8d^2}}{\frac{\pi R^2}{4d^2}} = \frac{R^2}{2}.
\end{aligned}$$

(c) With $R_1 = 2d$, the continuous assumption gives $\overline{E}^{R_1}_{cont}(d) = \overline{E}^{R=2d}_{cont}(d) = \frac{(2d)^2}{2} = 2d^2$. With $R_2 = \sqrt{20}d$, we get $\overline{E}^{R_2}_{cont}(d) = \frac{\sqrt{(20d)}^2}{2} = 10d^2$ which is a reasonable approximation for the exact discrete average energy $\overline{E}^{R_2}(d)$ previously computed!

(d) To transmit one extra bit, we have to double the number of points, i.e., we have to take a new number of points equal to $N_{new} = 2N(R)$. The relation $N(r) = \frac{pir^2}{4d^2}$ gives the radius R_{new} of the new constellation as $R_{new} = \sqrt{2}R$, i.e., we have to scale the radius by $\sqrt{2}$.

If $\frac{d}{\sigma} << 1$, the high order terms in the expression of the error probability are neglectible: we have $\Pr\{error\} \approx 4Q(\frac{d}{\sigma})$.

(e) To keep the error probability constant, d is set to be constant. Therefore, we have to scale the radius by $\sqrt{2}$ to transmit one extra point. The average energy $\overline{E}_{cont}^{R_{new}}(d)$ of the new constellation is then $\overline{E}_{cont}^{\sqrt{2}R}(d) = 2\frac{R^2}{2} = 2\overline{E}_{cont}^{R}(d)$, i.e., it is scaled by 2.

Exercise 1.11 For $H(z)$ for which the degree of $F(z)$ is less than the degree of $G(z)$, the (unique) partial fraction expansion can be written as,

$$H(z) = \frac{F(z)}{\prod_k (z - z_k)} = \sum_k \frac{\alpha_k}{z - z_k}. \tag{6.5}$$

for some coefficients α_k that we have now to determine.
Take an indice k_0 in the set of all the k's, the evaluation of the rational z-transform $(z - z_{k_0})H(z)$ in z_{k_0} is denoted $(z - z_{k_0})H(z)|_{z_{k_0}}$. We have,

$$H(z)(z - z_{k_0}) = \frac{F(z)}{\prod_{k:k\neq k_0}(z - z_k)},$$

such that

$$H(z)(z - z_{k_0})|_{z_{k_0}} = \frac{F(z_{k_0})}{\prod_{k:k\neq k_0}(z_{k_0} - z_k)} \tag{6.6}$$

$$= \frac{F(z_{k_0})}{G'(z_{k_0})}, \tag{6.7}$$

since $G'(z) = \sum_j \prod_{k:k\neq j}(z - z_k)$.
Eq. 6.7 combined with the evaluation of Eq. 6.5 in z_{z_0} leads to $\alpha_{k_0} = \frac{F(z_{k_0})}{G'(z_k)}$. Therefore,

$$H(z) = \sum_k \frac{F(z_k)}{G'(z_k)} \frac{1}{z - z_k} = \sum_k \frac{F(z_k)}{G'(z_k)} \frac{1}{z} \frac{1}{1 - \frac{z_k}{z}}.$$

Since $\frac{1}{z}\frac{1}{1 - z_k/z} = \sum_{j=0}^{\infty} {z_k}^j z^{-j-1}$, it is easy to see that the time sequence which possesses this z-transform $H(z)$ is causual with coefficients,

$$h_{n+1} = \sum_{k\geq 0} \frac{F(z_k)}{G'(z_k)} z_k^n.$$

Moreover, we can write $\frac{1}{z}\frac{1}{1 - z_k/z} = \frac{-1}{z_k}\frac{1}{1 - z/z_k} = -\sum_{j=0}^{\infty} {z_k}^{-j-1}\, z^j$ to obtain the corresponding anticausal time serie,

$$H(z) = \sum_k \frac{F(z_k)}{G'(z_k)} \frac{1}{1 - z_k} = \sum_{j\geq 0} \sum_k \frac{-(z_k)^{-j-1} F(z_k)}{G'(z_k)}\, z^j,$$

with coefficients,

$$\tilde{h}_n = \sum_{k\geq 0} \frac{-F(z_k)}{G'(z_k)} {z_k}^{-n-1}.$$

If the degree of $F(z)$ is larger, or equal to the degree of $G(z)$, a polynomial in z called $P(z)$ (with degree > 0 or equal to 0, respectively) will appear in the partial fraction expansion such that,

$$H(z) = P(z) + \sum_k \frac{\alpha_k}{z - z_k}.$$

Exercise 1.12

1. (a) With,

$$\begin{aligned}
\{x_i\}_{i\geq 0} &\leftrightarrow x(D) := \sum_{i=0}^{\infty} x_i D^i, \\
\{x_{i+1}\}_{i\geq 0} &\leftrightarrow x^{(1)}(D) := \sum_{i=0}^{\infty} x_{i+1} D^i, \\
\{x_{i+2}\}_{i\geq 0} &\leftrightarrow x^{(2)}(D) = \sum_{i=0}^{\infty} x_{i+2} D^i,
\end{aligned}$$

we can write,

$$\begin{aligned}
Dx^{(1)}(D) &= x(D) - x_0, \\
D^2 x^{(2)}(D) &= x(D) - x_0 - x_1 D.
\end{aligned}$$

1. (b) With $x'(D) := \sum_{i=1}^{\infty} i x_i D^{i-1}$, we have

$$\{i x_i\}_{i\geq 0} \leftrightarrow \sum_{i=0}^{\infty} i x_i D^i = D x'(D).$$

2. We could call 'exponential', the formal power sum,

$$\mathrm{e}^D := x_1(D) = \sum_{i=0}^{\infty} \frac{D^i}{i!},$$

and 'cosinus', the formal power sum,

$$\cos(D) := x_2(D) = \sum_{i=0}^{\infty} (-1)^i \frac{D^{2i}}{(2i)!}.$$

Those formal power sums have formal derivatives,

$$(\mathrm{e}^D)' = x_1'(D) = \sum_{i=1}^{\infty} i \frac{D^{i-1}}{i!} = \sum_{j=0}^{\infty} \frac{D^j}{j!} = x_1(D) = \mathrm{e}^D,$$

and,

$$\cos'(D) = x_2'(D) = \sum_{i=1}^{\infty} (-1)^i (2i) \frac{D^{2i-1}}{(2i)!} = -\sum_{i=1}^{\infty} (-1)^{i-1} \frac{D^{2i-1}}{(2i-1)!} =: -\sin(D),$$

defining the formal power sum 'sinus' being $\sin(D) = \sum_{i=1}^{\infty} (-1)^{i-1} \frac{D^{2i-1}}{(2i-1)!}$. The so-defined formal power sum $f(D)$ will have all formal properties of the corresponding function $f(z)$. By 'formal' properties, we mean properties formally coming from their series expansion.

3. *Expression for $a(D)$:*
For all $i \geq 0$, we have,

$$a_i + a_{i+1} = a_{i+2},$$

then

$$\sum_{i=0}^{\infty} a_{i+2} D^i = \sum_{i=0}^{\infty} (a_i + a_{i+1}) D^i = \sum_{i=0}^{\infty} a_i D^i + \sum_{i=0}^{\infty} a_{i+1} D^i,$$

which implies,

$$a(D) - a_0 - a_1 D = D^2 a(D) + D\Big(a(D) - a_0\Big).$$

Using $a_0 = a_1 = 1$ and observing that $1 - D - D^2$ is invertible since the constant term is non-zero, we get

$$a(D) = \frac{1}{1 - D - D^2}.$$

Let's find formally the four first coefficients :
A first method consist in identifying term by term the coefficient of the formal power sum. Formally $(1 - D - D^2) \cdot a(D) = 1$. Using $b_0 = 1, b_1 = b_2 = -1$ and $b_i = 0$ for $i \geq 3$, we can write $1 - D - D^2 = \sum_{=0}^{\infty} b_i D^i$. Therefore, by definition,

$$\Big(1 - D - D^2\Big) \cdot a(D) = \sum_{i=0}^{\infty} \sum_{l=0}^{i} b_l a_{i-l} D^i.$$

Then, for :

$$\begin{aligned}
i = 0 \quad & b_0 \cdot a_0 = b_0 \cdot 1 \\
i = 1 \quad & b_1 \cdot a_0 + b_0 \cdot a_1 = b_1 \cdot 1 + 1 \cdot 1 = 0 \\
i = 2 \quad & b_2 \cdot a_0 + b_1 \cdot a_1 + b_0 \cdot a_2 = -1 \cdot 1 + (-1) + a_2 = 0 \\
i = 3 \quad & b_3 a_0 + b_2 a_1 + b_1 a_2 + b_0 a_3 = -1 - 2_1 \cdot a_3 = 0
\end{aligned}$$

A second algebraic method consist in considering the long division.

$$\frac{1}{1 - D - D^2} = 1 + D + 2D^2 + 3D^3 + \dots$$

Formally, we found $a_0 = a_1 = 1, a_2 = 2, a_3 = 3$, which could also be found using a classical induction directly on the $\{a_i\}_i$.

Partial fraction expansion :
Since $1-D-D^2 = -\left(D+\frac{1+\sqrt{5}}{2}\right)\left(D+\frac{1-\sqrt{5}}{2}\right)$, we can write

$$a(D) = \frac{1}{1-D-D^2} = \frac{\alpha}{D+\frac{1+\sqrt{5}}{2}} + \frac{\beta}{D+\frac{1-\sqrt{5}}{2}}.$$

Using,

$$[a(D)(\frac{1+\sqrt{5}}{2}+D)](-\frac{1+\sqrt{5}}{2}) = \alpha = \frac{\sqrt{5}}{5},$$

and,

$$[a(D)(\frac{1-\sqrt{5}}{2}+D)](-\frac{1-\sqrt{5}}{2}) = \beta = -\frac{\sqrt{5}}{5},$$

we get the partial fraction expansion,

$$a(D) = \frac{\sqrt{5}}{5}\Big(\frac{1}{\frac{1+\sqrt{5}}{2}+D} - \frac{1}{\frac{1-\sqrt{5}}{2}+D}\Big).$$

This procedure was clearly defined in a purely formal way since we just used algebraic operations for formal power sum.

Expression of the coefficients a_i :
Since formally, for all $\gamma \in F$,

$$\frac{1}{1+\gamma D} = \sum_{i=0}^{\infty}(-1)^i(\gamma D)^i,$$

we can write,

$$\begin{aligned} a(D) &= \frac{2}{\sqrt{5}+5}\cdot\frac{1}{1+\frac{2}{1+\sqrt{5}}D} - \frac{2}{\sqrt{5}-5}\cdot\frac{1}{1+\frac{2}{1-\sqrt{5}}D} \\ &= \frac{2}{\sqrt{5}+5}\cdot\sum_{i=0}^{\infty}\Big(\frac{-2}{1+\sqrt{5}}D\Big)^i - \frac{2}{\sqrt{5}-5}\cdot\sum_{i=0}^{\infty}\Big(\frac{-2}{1-\sqrt{5}}D\Big)^i \\ &= \sum_{i=0}^{\infty}\Big[\frac{2}{\sqrt{5}+5}\cdot\Big(\frac{-2}{1+\sqrt{5}}\Big)^i - \frac{2}{\sqrt{5}-5}\cdot\Big(\frac{-2}{1-\sqrt{5}}\Big)^i\Big]D^i, \end{aligned}$$

i.e., for all $i \geq 0$,

$$a_i = \frac{2}{\sqrt{5}+5}\cdot\Big(\frac{-2}{1+\sqrt{5}}\Big)^i - \frac{2}{\sqrt{5}-5}\cdot\Big(\frac{-2}{1-\sqrt{5}}\Big)^i.$$

4. With $(i+1)a_{i+1} = 3a_i + 1, (i \geq 0; a_0 = 1)$, we have now,

$$\sum_{i=0}^{\infty}(i+1)a_{i+1}D^i = 3\sum_{i=0}^{\infty}a_iD^i + \sum_{i=0}^{\infty}D^i,$$

i.e.,

$$a'(D) = 3 \cdot a(D) + \frac{1}{1-D}.$$

We can use the real topology and analytical operations to try to find the components $\{a_i\}_i$. Consider the differential equation,

$$f'(x) - 3f(x) = \frac{1}{1-x}.$$

We get the solution $f(x) = c \cdot e^{3x}$ for the homogenous system $f' - 3f = 0$. By variation of the constant, a function $c(x) \cdot e^{3x}$, solution of $f'(x) - 3f(x) = \frac{1}{1-x}$, can be found to be $c(x) = \int \frac{e^{-3x}}{1-x} dx$. Unfortunatly this integral is difficult to solve using simple mathematical tools. However, the reader will observe that the set of functions $\{x \mapsto (A + c(x))e^{3x}\}_{A \in \mathbb{R}}$ satisfying the differential equation permit us to determine the real formal power sum $a(D)$ by finding its components on $\mathbb{R}$.

5. Using $x(D) = \sum_{i=0}^{\infty} x_i D^i$ and $y(D) = \sum_{i=0}^{\infty} y_i D^i$, we have $y(x(D)) = \sum_{i=0}^{\infty} y_i (x(D))^i = \sum_{i=0}^{\infty} y_i (x_0 + D\sum_{j=0}^{\infty} x_{j+1} D^j)^i = \sum_{i=0}^{\infty} y_i [x_0^i + P^{(i)}(D)]$, where $P^{(i)}(D)$ is a polynomial of degree $i \times \deg(P(D))$ without degree zero coefficients. It can be splitted in two terms: the first is $\sum_{i=0}^{\infty} y_i x_0^i = x_0^i \sum_{i=0}^{\infty} y_i$ and the second $\sum_{i=0}^{\infty} y_i P^{(i)}(D)$ is always well-defined.
(i) If $x_0 = 0$, then the computation of any coefficient of $y(x(D))$ requires a *finite* number of operations, i.e., we will be able to compute all the coefficients of $y(x(D))$.
(ii) If $x_0 \neq 0$, the constant coefficient requires already an infinite number of operations for being computed. The same statement can be made for the other coefficients. Therefore $g(x(D))$ will not be defined in general, except if $x(D)$ is a finite formal power sum, i.e., a polynomial.
Conclusion : The composition $y(x(D))$ is defined iff

$$x_0 = 0$$

or

$$y(D) \text{ is a polynomial.}$$

Then, $e^{e^D - 1}$ with $x_0 = 0$ is a well-defined formal power sum whereas e^{e^D} is not.

Exercise 1.13

1. The inverse $\frac{1}{x(D)}$ exists if and only if x_0 is invertible. Since for our case $x_0 = 1$, $\frac{1}{x(D)}$ exists. The composition $x(y(D))$ exists if either $y_0 = 0$ or $x(D)$ is a polynomial. In our case $y_0 = 0$, so that $x(y(D))$ exists. We

get the first few coefficients of $z(D) := \frac{1}{x(D)}$ by using the equations

$$\begin{aligned} z_0 &= \frac{1}{x_0}, \\ z_i &= -\frac{1}{x_0}\sum_{j=0}^{i-1} z_j x_{i-j}, \quad i \geq 1. \end{aligned}$$

In a similar way we get the first few coefficients of $x(y(D))$ by determining the lower order terms of $x_0 + x_1 y(D) + x_2 y^2(D) \cdots$. The result is $x_0 = 1$, $x_1 = -1$, $x_2 = \frac{1}{2}$, $y_0 = 1$, $y_1 = 1$, $y_2 = 0$, $y_0 = 1$, $y_1 = 1$, $y_2 = 0$.

2. We get

$$x'(D) = \left(\sum_{i=0}^{\infty} \frac{1}{i!} D^i\right)' = \sum_{i=0}^{\infty} \frac{i}{i!} D^{i-1} = \sum_{i=1}^{\infty} \frac{1}{(i-1)!} D^{i-1} = \sum_{i=0}^{\infty} \frac{1}{i!} D^i = x(D),$$

and

$$y'(D) = \left(-\sum_{i=1}^{\infty} \frac{(-1)^i}{i} D^i\right)' = \sum_{i=1}^{\infty} \frac{(-1)^{i-1} i}{i} D^{i-1} = \sum_{i=0}^{\infty} (-1)^i D^i = \frac{1}{1+D}.$$

3. We find $f(D) = e^D$ and $g(D) = \ln(1+D)$.
4. Using $\frac{1}{f(D)} = e^{-D}$ we get $\frac{1}{x(D)} = \sum_{i=0}^{\infty} \frac{(-1)^i}{i!} D^i$. Further, since $f(g(D)) = 1 + D$, we get $x(y(D)) = 1 + D$.

Exercise 1.14

Greatest common divisor of 1573 and 308 :

We have,

$$\begin{aligned} 1573 &= 308 \times 5 + 33, \\ 308 &= 33 \times 9 + 11, \\ 33 &= 11 \times 3, \end{aligned}$$

so that,

$$\gcd(1573, 308) = \gcd(308, 33) = \gcd(33, 11) = 11.$$

Extension of this algorithm :

We can easily extend this algorithm to find the Bezout equality by computing recursively,

$$\begin{aligned} 11 &= 308 - 33 \times 9) \\ &= 308 - (1573 - 308 \times 5) \times 9 \\ &= a1573 + b308, \end{aligned}$$

with $a = -9$ and $b = 46$. We could also show that the set of all (a,b) which verify the equality is $\{-9 - c308, 46 + c1573 : c \in \mathbb{Z}\}$.

Euclidian algorithm for polynomials :
Using the Euclidian algorithm, we can calculate the greates common divisor of two polynomials. Recursive divisions occur as long as the remainder remains a non-constant polynomial. If this constant polynomial is zero, then the polynomial of smallest degree divides the polynomial of greater degree. If this constant polynomial is a non-zero one, then the two polynomial are relatively prime.
Therefore using Fig. 6.5, we get,

$$\begin{array}{r|l} x^4 - x^3 + x - 1 & x^2 - x + 1 \\ -(x^4 - x^3 + x^2) & \hline x^2 - 1 \\ \hline -x^2 + x - 1 & \\ -(-x^2 + x - 1) & \\ \hline 0 & \end{array} \qquad \begin{array}{r|l} x^4 - x^2 + x - 1 & x^3 - x^2 + 1 \\ -(x^4 - x^3 + x) & \hline x + 1 \\ \hline x^3 - x^2 - 1 & \\ -(x^3 - x^2 + 1) & \\ \hline -2 & \end{array}$$

Figure 6.5: Euclidian Division in the Polynomial Ring.

$$\begin{aligned} \gcd\left(x^4 - x^3 + x - 1, x^2 - x + 1\right) &= x^2 - x + 1, \\ \gcd\left(x^4 - x^2 + x - 1, x^3 - x^2 + 1\right) &= \gcd\left(x^3 - x^2 + 1, -2\right) = 1. \end{aligned}$$

Exercise 1.15

If poss, try to check this ex. more carefully

For a fixed $m \geq 0$, we have,

$$\begin{aligned}
\sum_{n=0}^{\infty} f(n,m)x^n &= \sum_{n=0}^{\infty}\sum_{k\geq 0}\binom{n+k}{m+2k}\binom{2k}{k}\frac{(-1)^k}{k+1}x^n \\
&= \sum_{k=0}^{\infty}\sum_{n=0}^{\infty}\binom{n+k}{m+2k}\binom{2k}{k}\frac{(-1)^k}{k+1}x^n \\
&= \sum_{k=0}^{\infty}\binom{2k}{k}\frac{(-1)^k}{k+1}\sum_{n=0}^{\infty}\binom{n+k}{m+2k}x^n \\
&= \sum_{k=0}^{\infty}\binom{2k}{k}\frac{(-1)^k}{k+1}\sum_{j=k}^{\infty}\binom{j}{m+2k}x^{j-k} \\
&= \sum_{k=0}^{\infty}\binom{2k}{k}\frac{(-1)^k}{k+1}\cdot x^{-k}\cdot\sum_{j=0}^{\infty}\binom{j}{m+2k}x^{j} \\
&= \sum_{k=0}^{\infty}\binom{2k}{k}\frac{(-1)^k}{k+1}\cdot x^{-k}\cdot\frac{x^{m+2k}}{(1-x)^{m+2k+1}} \\
&= \sum_{k=0}^{\infty}\binom{2k}{k}\frac{1}{k+1}\Big(\frac{-x}{(1-x)^2}\Big)^k\frac{x^m}{(1-x)^{m+1}} \\
&= \Big(\frac{(1-x)^2}{-2x}\cdot\Big(1-\sqrt{\frac{(1+x)^2}{(1-x)^2}}\Big)\Big)\frac{x^m}{(1-x)^{m+1}}.
\end{aligned}$$

The square root makes that we have 2 distinct cases:

– Case $x > 1$ or $x < -1$:

$$\sum_{n=0}^{\infty} f(n,m)x^n = \frac{x-1}{x}\frac{x^m}{(1-x)^{m+1}}$$

– Case $-1 < x < 1$:

$$\sum_{n=0}^{\infty} f(n,m)x^n = (1-x)\frac{x^m}{(1-x)^{m+1}}$$

Therefore, taking $0 < x < 1$, we can write

$$\begin{aligned}
\sum_{m=0}^{\infty}\sum_{n=0}^{\infty} f(n,m)x^n y^m &= (1-x)\sum_{m=0}^{\infty}\frac{x^m}{(1-x)^{m+1}}y^m \\
&= (1-x)\frac{1}{1-x-xy}
\end{aligned}$$

for all (x,y) such $0 \leq \frac{xy}{1-x} < 1$ (Remark that there exists $y < \frac{1-x}{x}$).
It remains simply to use the Taylor expansion in y and then in x to determine the coefficients of the function which are the exact estimates of the functions

$f(n,m)$. It can be done simply as in the following:

$$\begin{aligned}
\sum_{m=0}^{\infty}\sum_{n=0}^{\infty} f(n,m)x^n y^m &= \frac{1-x}{1-x-xy} \\
&= \frac{1}{1-\frac{xy}{1-x}} \\
&= \sum_{i=0}^{\infty} y^i x^i (1-x)^{-i} \\
&= \sum_{i=0}^{\infty} y^i x^i \left(1+\sum_{p=1}^{\infty}\frac{(-i)(-i-1)\cdots(-i-p+1)}{p!}x^p\right) \\
&= \sum_{i=0}^{\infty} y^i x^i \left(\sum_{p=0}^{\infty}(-1)^p\binom{i+p-1}{p}x^p\right) \\
&= \sum_{i=0}^{\infty}\sum_{p=0}^{\infty}(-1)^p\binom{i+p-1}{p}x^{i+p}y^i \\
&= \sum_{i=0}^{\infty}\sum_{n=i}^{\infty}(-1)^{n-i}\binom{n-1}{n-i}x^n y^i \\
&= \sum_{m=0}^{\infty}\sum_{n=0}^{\infty}(-1)^{n+m}\binom{n-1}{n-m}x^n y^m.
\end{aligned}$$

We get, $f(n,m) = (-1)^{n+m}\binom{n-1}{n-m}$, for all couples (m,n).

7

SOLUTIONS OF THE EXERCISES - 2

Exercise 2.1
Consider the shift register shown in Fig. 7.1 which describes all the states of a finite-state machine. With a $|\mathcal{X}|$-ary alphabet, the size of the state space is $|\mathcal{X}|^{L-1}$ and there are $|\mathcal{X}| \times |\mathcal{X}|^{L-1} = |\mathcal{X}|^{L}$ edges per trellis section. In binary, we get 2^{L-1} states and 2^{L} edges.

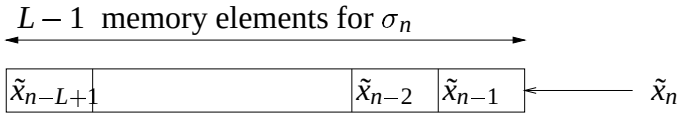


Figure 7.1: Finite-State Machine: State of the Markov Chain.

Exercise 2.2
The transmitted symbols x_n are i.i.d. random variables, taking on $+1$ and -1 equally likely, and the received symbols are given by

$$y_n = \prod_{i=1}^{n} x_i + z_n, \;\; n = 1, \cdots, N,$$

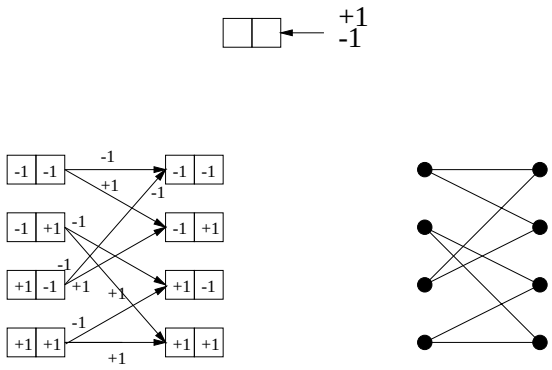


Figure 7.2: Shift Register for $L = 3$.

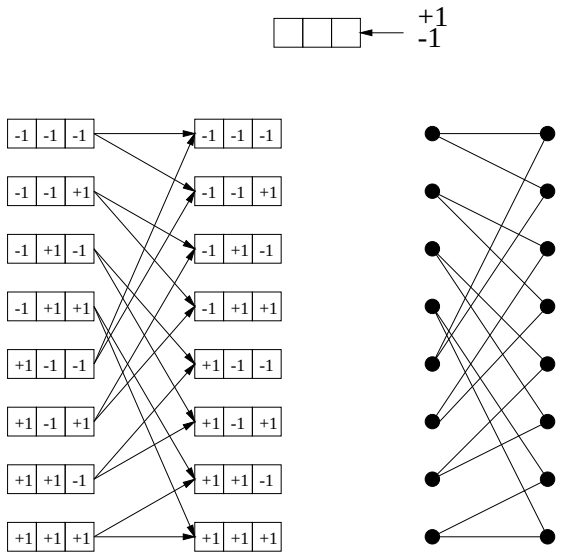


Figure 7.3: Shift Register for $L = 4$.

where z_n is an i.i.d. sequence of random variables with density $p(z)$. We have,

$$\begin{aligned}
p(y_1,\cdots,y_N|x_1,\cdots,x_N) &\overset{(a)}{=} p(y_1-x_1,\cdots,y_j-\prod_{i=1}^{j}x_i,\cdots,y_N-\prod_{i=1}^{N}x_i \mid x_1,\cdots,x_N)\\
&\overset{(b)}{=} \prod_{j=1}^{N} p(y_j-\prod_{i=1}^{j}x_i \mid x_1,\cdots,x_N)\\
&\overset{(c)}{=} \prod_{j=1}^{N} p(y_j-\prod_{i=1}^{j}x_i \mid x_1,\cdots,x_j)\\
&\overset{(d)}{=} \prod_{j=1}^{N} p(y_j-\sigma_j x_j \mid \sigma_j,x_j)\\
&\overset{(e)}{=:} \prod_{n=1}^{N} f(y_n;x_n;\sigma_n),
\end{aligned}$$

where (a) comes from conditioning, (b) comes from the fact that z_n is an i.i.d. sequence of random variables, (c) is valid since the output is independent of the future inputs, (d) is obtained with $\sigma_j=\prod_{i=1}^{j-1}x_i$ after having noticed that the output only depends on the *product* ($\prod_{i=1}^{j-1}\in\{-1,+1\}$) of the x_i and (e) is written by defining $f(j_j;x_j;\sigma_j)$ being $p(y_j-\sigma_j x_j \mid \sigma_j,x_j)$. A suitable *state* (which has a small state space) is therefore $\sigma_n=\prod_{i=1}^{n-1}x_i$. The Viterbi algorithm will find, after Bayes inversion, the sequence,

$$\text{argmax}_{x_1,\cdots,x_N}\, p(y_1,\cdots,y_N) = \text{argmax}_{x_1,\cdots,x_N}\prod_{n=1}^{N} p_{Z_n}(y_n-\sigma_n x_n),$$

as shown in Fig. 7.4 for the example of the exercise where $N=4$ and where

$$p_Z(z) := \begin{cases} \frac{2-|z|}{4}, & |z|\le 2,\\ 0, & \text{otherwise.}\end{cases}$$

Assuming that the received sequence is

$$y_1=-0.1, y_2=0.5, y_3=0.9, y_4=-0.2,$$

the Viterbi algorithm to this case to find the most likely transmitted sequence $(\tilde{x}_1,\tilde{x}_2,\tilde{x}_3,\tilde{x}_4)=(-1\ -1\ +1\ -1)$. The complexity of the Viterbi algorithm is roughly $|\mathcal{X}|^{L-1}$, (here $4\times 2^1=8$). The complexity is around $3|\mathcal{X}|^{L-1}$ for the BCJR.

Exercise 2.3
We have the discrete-time channel of Fig. 7.5. The states of the channel permit us to see it as a finite-state machine.

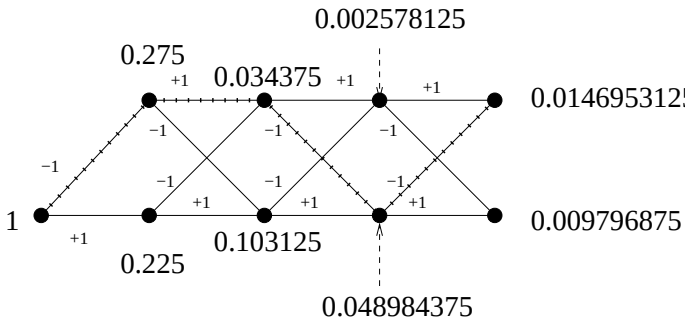


Figure 7.4: Binary trellis with $L = 2$ and $N = 4$.

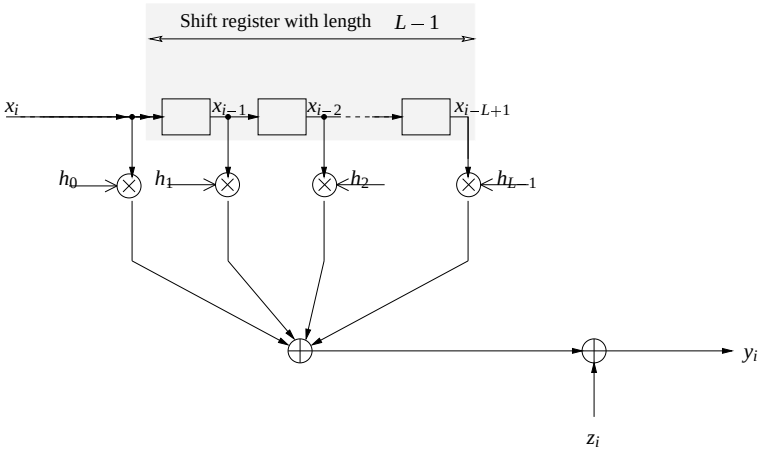


Figure 7.5: Channel Model: additive i.i.d. Gaussian Noise, L Taps.

We consider sequences of N bits with $L << N$. Assuming the channel initial state is known, i.e., $\tilde{x}_{-L+1}, \tilde{x}_{-L+2}, \cdots, \tilde{x}_{-1}$ are known. We have,

$$\begin{aligned} p(y_0, \cdots, y_{N-1} | \tilde{x}_0, \cdots, \tilde{x}_{N-1}) &= p_{\{Y_i\}_i | \{\tilde{X}_i\}_i}(y_0, \cdots, y_{N-1} | \tilde{x}_0, \cdots, \tilde{x}_{N-1}) \\ &= p_{\{Y_i\}_i | \{\tilde{X}_i\}_i}\Big(\{y_i\}_{0 \le i \le N-1} | \{\tilde{x}_i\}_{0 \le i \le N-1}\Big) \\ &= p_{\{Z_i\}_i | \{\tilde{X}_i\}_i}\Big(\{y_i - \sum_{n=0}^{L-1} h_n \tilde{x}_{i-n}\}_{0 \le i \le N-1} | \{\tilde{x}_i\}_{0 \le i \le N-1}\Big) \\ &= p_{\{Z_i\}_i}\Big(\{y_i - \sum_{n=0}^{L-1} h_n \tilde{x}_{i-n}\}_{0 \le i \le N-1}\Big) \\ &= \prod_{0 \le i \le N-1} p_{Z_i}(y_i - \sum_{n=0}^{L-1} h_n \tilde{x}_{i-n}), \end{aligned}$$

where $p_{Z_i}(y_i - \sum_{n=0}^{L-1} h_n \tilde{x}_{i-n}) = \frac{1}{2\pi\sigma^2} \exp(-\frac{||y_i - \sum_{n=0}^{L-1} h_n \tilde{x}_{i-n}||^2}{2\sigma^2})$ since the Z_i are i.i.d. circularly-symmetric complex-valued Gaussian random variables. Skipping the indices, we get,

$$\log p(y_0, \cdots, y_{N-1} | \tilde{x}_0, \cdots, \tilde{x}_{N-1}) = \sum_{i=0}^{N-1} \log p(y_i - \sum_{n=0}^{L-1} h_n \tilde{x}_{i-n}),$$

and, using the notations $\sigma_i = (\tilde{x}_{i-L+1}, \tilde{x}_{i-L+2}, \cdots, \tilde{x}_{i-1})$ and $m(y_i; \tilde{x}_i; \sigma_i) = \log p(y_i - \sum_{n=0}^{L-1} h_n \tilde{x}_{i-n})$, we can write,

$$\max_{\tilde{x}_0, \cdots, \tilde{x}_{N-1}} p(y_0, \cdots, y_{N-1} | \tilde{x}_0, \cdots, \tilde{x}_{N-1}) = \max_{\tilde{x}_0, \cdots, \tilde{x}_{N-1}} \sum_{i=0}^{N-1} m(y_i; \tilde{x}_i; \sigma_i).$$

We have seen in class that an efficient algorithm to accomplish this task (maximization on the sequence) is called *Viterbi* algorithm.

Exercise 2.4
A trellis section for an (inter-symbol interference) ISI channel should have $|\mathcal{X}|^{L-1}$ states and $|\mathcal{X}|$ outgoing edges per state. It follows that (a) can not be such a trellis section, but (b), (c) and (d) fulfill this criterion. From our homework we know that the section in (b) is the trellis section for the case $|\mathcal{X}| = 2$ and $L = 3$ and, similarly, we see that the figure in (c) corresponds to the case $|\mathcal{X}| = 4$ and $L = 2$. The figure in (d) does not correspond to a valid trellis section since it does not have the required "butterfly" structure.

Exercise 2.5
The preliminaries have been reviewed in class.

Exercise 2.6
The first method used in the Internet was the *distance vector* routing which

does not have a detailed global view of the network. Now a topology database is elaborated in each router using *link state* updates. Every router has to perform a kind of Viterbi algorithm called Dijkstra algorithm. This algorithm is implemented in most of the recent routers using TCP/IP. (See networking lecture for more informations).
Assuming a student from EPFL located in S in Fig. 7.6 wants to go quickly to to eat some tampas on the ramblas in E. He will intuitively perform the following example of a shortest path algorithm before taking his car.

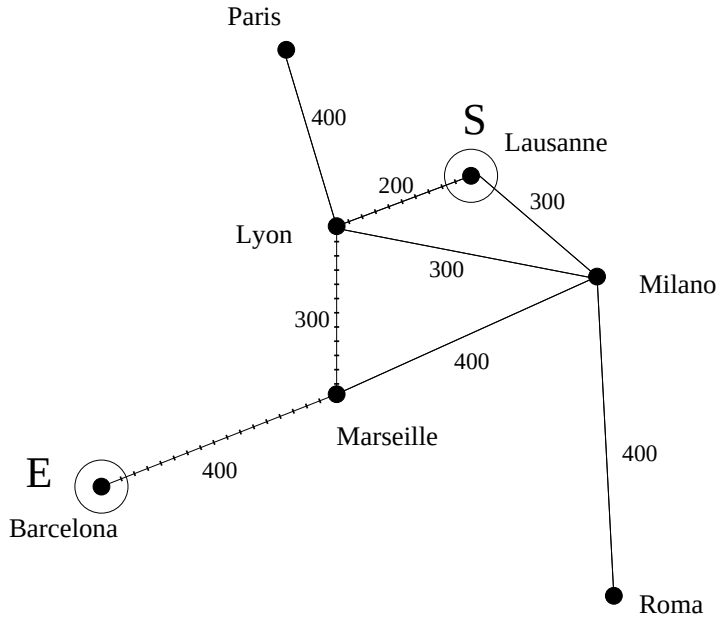


Figure 7.6: Graph with Cities as Nodes and Routes as Edges.

A time $t = 1$, Milano is at 300km using edge Lausanne-Milano. OK.
A time $t = 2$, Lyon is at 200km using edge Lausanne-Lyon. OK.
A time $t = 3$, Lyon is at 600km using edge Milano-Lyon from state Milano. This path is not the shortest one. Path can be rejected since the surviving path at the edge Lyon has to be the shortest one.
A time $t = 4$, Marseille is at 500km using edge Lyon-Marseille from state Lyon. OK.
A time $t = 5$, Marseille is at 700km using edge Milano-Marseille from state Milano. Path is rejected.
A time $t = 6$, Barcelona is a 900km using edge Marseille-Barcelona from state Marseille with cumulative value 500km. OK.
The rest of the computations will not affect the value of the shortest path.

The shortest path algorithm indicates us that the node E is at distance 900 of the source node S.

Exercise 2.7
Given $i \in \{1, \cdots, N\}$ and the corresponding couple (x_{i-1}, x_i). The couple is now chosen and fixed. The APP estimator for the algorithm is,

$$\begin{aligned}
(\tilde{x}_{i-1}, \tilde{x}_i) &= \text{argmax}_{(x_{i-1},x_i)}\, p(x_{i-1}, x_i | y_0, y_1, \cdots, y_N) \\
&\overset{(a)}{=} \text{argmax}_{(x_{i-1},x_i)}\, p(\underbrace{y_0, y_1, \cdots, y_N}_{\vec{y}} | x_{i-1}, x_i) \\
&= \text{argmax}_{(x_{i-1},x_i)} \sum_{\vec{x}\ s.t.\ x_{i-1},x_i} p(\vec{y}|\vec{x}) \\
&\overset{(b)}{=} \text{argmax}_{(x_{i-1},x_i)} \sum_{\vec{x}\ s.t.\ x_{i-1},x_i} \prod_{j=0}^{N} p(y_j - \sum_{\ell=0}^{L-1} h_\ell x_{j-\ell} | \underbrace{(x_{j-L+1}, \cdots, x_{j-2}), x_{j-1}}_{\sigma_j}, x_j) \\
&\overset{(c)}{=} \text{argmin}_{(x_{i-1},x_i)} \sum_{\vec{x}\ s.t.\ x_{i-1},x_i} ||y_j - \sum_{\ell=0}^{L-1} h_\ell x_{j-\ell}||
\end{aligned}$$

where (a) is induced by Bayes and the equal priors, (b) comes from the noise sequence of i.i.d. random variables and (c) comes from the Gaussian characteristics of the noise. Briefly, the algorithm will have approximately the same complexity as the BCJR and will be similar regarding the forward and backwar recursions. The distinction will be in the γ-estimation: here the estimate for the initial couple (x_{i-1}, x_i) will be obtained by additioning the probabilities of all the states 9at a fixed time) in which the couple appears.

Exercise 2.8
Recall that, by convention, the *region of convergence* of a rational filter $H(z) = \frac{P(z^{-1})}{Q(z^{-1})} = \sum_{-\infty}^{\infty} h_n z^{-n}$ is the set of all complex of magnitude larger than the magnitude of the largest magnitude poles. This set contains ∞. The filter $H(z)$ is

- *causal* if the system is non-anticipative, i.e., if $h_n = 0$ for all $n < 0$. This justifies the convention that the region of convergence must be outside the outermost pole.
- *stable* if a bounded input leads to a bounded output which can be proven to be equivalent to all poles are inside the unit circle.
- *minimum phase* if all zeros and poles are inside the unit circle.

1. *Causality :*
   The point ∞ is not a zero, i.e., $\deg_{z^{-1}}\{P(z^{-1})\} \leq \deg_{z^{-1}}\{Q(z^{-1})\}$.

Therefore, using the partial fractions expansion over the complex plane, we can first write $H(z)$ as $H(z) = g_0 + \sum_{n=0}^{\infty} \frac{g_n z^{-1}}{(1-c_n z^{-1})^{k_n}}$. The geometric sum expansion of all partial fractions will secondly clearly lead to a causal extension of $H(z)$ of the form $H(z) = \sum_0^\infty h_n z^{-n}$.
Stability :
By definition, all poles are in the unit circle.
Conclusion :
A minimum phase filter is a causal and stable filter.

2. The inverse of $H(z)$ is $H^{-1} = \frac{Q(z^{-1})}{P(z^{-1})}$. It is again a minimum phase filter which is stable and causal.

3. We have

$$|H(\mathrm{e}^{2\pi jf})| = \frac{|\mathrm{e}^{-2\pi jf} - a^*|}{|1 - a\mathrm{e}^{-2\pi jf}|} = \sqrt{\frac{1 - 2\mathcal{R}e(a\mathrm{e}^{-2\pi jf}) + |a|^2}{1 - 2\mathcal{R}e(a\mathrm{e}^{-2\pi jf}) + |a|^2}} = 1.$$

4. Consider the all-pass filter $H_c(z) = \frac{z^{-1}-c^*}{1-cz^{-1}}$. Its inverse $H_c^{-1}(z) = \frac{1-cz^{-1}}{z^{-1}-c^*}$ is also a all-pass filter. Observe now that the filter $H(z)H_c^{-1}(z)$ derived from H(z) has same frequency response as $H(z)$. But, since $H(z)H_c^{-1}(z) = H_1(z)(1 - cz^{-1})$, the new filter having a new zero in c will no longer have any zeros or poles outside the unit circle.

   This approach can be easily generalized for a filter $H(z)$ with n zeros $c_1, c_2, \cdots, c_n$ outside the unit circle. A minimum phase filter with equal frequency can be derived from $H(z)$ by simply multiplying $H(z)$ by the corresponding product $\tilde{H}_{\mathrm{ap}}(z) = \prod_{i=1}^{n} H_{c_i}^{-1}(z)$, which is an all-pass filter. Denote $\tilde{H}_{\min}(z)$ this minimum phase filter, it is obtained as,

$$H_{\min}(z) = H(z)\tilde{H}_{\mathrm{ap}}(z).$$

   As asked in the exercise, notice that this equality can be written $H(z) = H_{\min}(z)H_{\mathrm{ap}}(z)$ where $H_{\mathrm{ap}} = [\tilde{H}_{\mathrm{ap}}(z)]^{-1}$ is also an all-pass filter.

5. We have

$$h_0 = H(\infty) = \lim_{z\to\infty} H(z).$$

   But $\lim_{z\to\infty} H_{\min}(z) = h_{\min,0}$ since a minimum-phase filter is causal. Morevore $\lim_{z\to\infty} |\tilde{H}_{\mathrm{ap}}(z)| = \prod_{i=1}^{n} |c_i|$ since $\tilde{H}_{\mathrm{ap}}(z)$ is a product of factor of the type $H_{c_i}^{-1}(z)$. Both limits exists and the limit of the product $H_{\min}(z)H_{\mathrm{ap}}(z) = H_{\min}(z)[\tilde{H}_{\mathrm{ap}}(z)]^{-1}$ is therefore the product $|h_0| = |h_{\min,0}| \times [\prod_{i=1}^{n} |c_i|]^{-1} < 1 \times |h_{\min,0}|$ since, for all i, $|c_i| > 1$. I.e., we get,

$$|h_0| < |h_{\min,0}|,$$

   in other words, a minimum phase filter $H_{\min}(z)$ derived from a filter $H(z)$ *maximizes* the partial energy term corresponding to $k = 0$.

Exercise 2.9

1. We have

$$\begin{aligned}
\mathcal{R}_w(k) &= \mathbb{E}[w_n w^*_{n-k}] \\
&= \mathbb{E}[\sum_{m\geq 0} f_m x_{n-m} w_n \sum_{i\geq 0} f_i^* x^*_{n-k-i}] \\
&= \mathbb{E}[\sum_{m\geq 0}\sum_{i\geq 0} f_m f_i^* x_{n-m} x^*_{n-(k+i)}] \\
&= \mathbb{E}[\sum_{m\geq 0}\sum_{l\geq k} f_m f^*_{l-k} x_{n-m} x^*_{n-l}] \\
&= \sum_{m\geq 0}\sum_{l\geq k} f_m f^*_{l-k} \mathbb{E}[x_{n-m} x^*_{n-m-(l-m)}] \\
&= \sum_{l\geq k} f^*_{l-k} \sum_{m\geq 0} f_m \mathbb{E}[x_{n-m} x^*_{n-m-(l-m)}] \\
&= \sum_{l\geq k} f^*_{l-k} \sum_{m\geq 0} f_m \mathcal{R}_x(l-m).
\end{aligned}$$

2. Assume first that f fulfils the equations (A.4). All the terms in the double sum are equal to zero so that

$$\forall k \geq 1 \quad \mathcal{R}_w(k) = 0.$$

3. Since $\mathcal{R}_w(k) = \mathcal{R}^*{}_w(-k)$, i.e., both quantities are conjugate symmetric, the previous work show that

$$\forall k \geq 1 \quad \mathcal{R}_w(-k) = 0.$$

   Globally this implies that $\mathcal{R}_w(k) = 0$ for $k \neq 0$. Therefore w is white, i.e., f is a whitening filter.

Exercise 2.10
The equivalent discrete time channel model is

$$y_n = \sum_k \mathcal{R}_g(k) x_{n-k} + z_n,$$

where z_n is a complex valued circularly symmetric Gaussian process with $\mathcal{R}_z(k) = N_0 \mathcal{R}_g(k)$. The z-transform leads to

$$Y(z) = \mathcal{S}_g(z) X(z) + Z(z).$$

Assume we filter this received signal through some filter $F(z)$.

1. To eliminate the intersymbol interference completely, we have to choose $F(z) = \frac{1}{\mathcal{S}_g(z)}$ (zero forcing criterion).

2. At the output of the filter, the signal is then $F(z)Y(z) = X(z) + F(z)Z(z)$ and the power spectral density of the noise $\tilde{Z}(z) := F(z)Z(z)$ is

$$\begin{aligned} S_{\tilde{z}} &= N_0 \mathcal{S}_g(z) \cdot F(z) \cdot F^*(\frac{1}{z^*}) \\ &= N_0 \mathcal{S}_g(z) F(z) \cdot F^*(\frac{1}{z^*}) \\ &= N_0 \mathcal{S}_g(z) \frac{1}{\mathcal{S}_g(z)} \frac{1}{\mathcal{S}_g^*(\frac{1}{z^*})} \\ &= N_0 \mathcal{S}_g(z) \frac{1}{\mathcal{S}_g(z)} \frac{1}{\mathcal{S}_g(\frac{1}{z})} \\ &= \frac{N_0}{\mathcal{S}_g(z)}. \end{aligned}$$

3. The noise power for this design criterion is then

$$\begin{aligned} \epsilon^2_{\text{LE-ZF}} &:= \mathbb{E}[|\tilde{z}_n|^2] = \mathcal{R}_{\tilde{Z}}(0) \\ &= \int_{-\frac{1}{2}}^{\frac{1}{2}} \frac{N_0}{\mathcal{S}_g(\mathrm{e}^{2\pi jf})} \mathrm{d}f > \int_{-\frac{1}{2}}^{\frac{1}{2}} \frac{N_0}{\mathcal{S}_g(\mathrm{e}^{2\pi jf}) + N_0} \mathrm{d}f =: \epsilon^2_{\text{LE-MMSE}}, \end{aligned}$$

using Equation 2.8.

Exercise 2.11

We use Jensen's inequality - concavity of the function ln in (a) - to write,

$$\begin{aligned} \epsilon_{\text{DFE-ZF}} &:= N_0 \exp\left\{ \int_{-\frac{1}{2}}^{\frac{1}{2}} \ln\Big(\frac{1}{\mathcal{S}_g(e^{2\pi jf})}\Big) df \right\} \\ &\overset{(a)}{\leq} N_0 \exp\left\{ \ln\left[\int_{-\frac{1}{2}}^{\frac{1}{2}} \Big(\frac{1}{\mathcal{S}_g(e^{2\pi jf})}\Big) df \right] \right\} \\ &= \int_{-\frac{1}{2}}^{\frac{1}{2}} \frac{N_0}{\mathcal{S}_g(\mathrm{e}^{2\pi jf})} \mathrm{d}f =: \epsilon_{\text{LE-ZF}}. \end{aligned}$$

Exercise 2.12

In the naive precoding scheme discussed in class, the transmitted signal is equal to $\tilde{X}(z) = \frac{X(z)}{B_0(z)}$ with $B_0(z) = \frac{\mathcal{S}_g(z)}{A_g}$. Assuming that $\mathcal{R}_x(k) = \delta(k)$, we get $\mathcal{S}_{\tilde{x}}(z) = \frac{\mathcal{S}_g^+(z)}{A_g} \frac{\mathcal{S}_g^{+*}(\frac{1}{z^*})}{A_g^*}$. Define $H(z)$ being the filter $H(z) = \frac{A_g}{\mathcal{S}_g^+(z)}$. $H(z)$ is the inverse of the monic filter $\frac{\mathcal{S}_g^+(z)}{A_g}$ and is itself monich and causal. We have,

$$\begin{aligned} \mathcal{R}_{\tilde{x}}(0) &= \int_{-\frac{1}{2}}^{\frac{1}{2}} |H(\mathrm{e}^{2\pi jf})|^2 \mathrm{d}f \\ &= \sum_{k=0}^{\infty} |h_k|^2, \end{aligned}$$

using Parseval and the causality property of H(z).Then

$$\mathcal{R}_{\tilde{x}}(0) \quad = \quad 1 + \sum_{k=1}^{\infty} |h_k|^2 \geq 1,$$

since the filter is monic.

8

SOLUTIONS OF THE EXERCISES - 3

Exercise 3.1
With,

$$\sum_{i=0}^{\infty} g_i D^i = G(D) := G_1(D) + G_2(D) = \sum_{i=0}^{\infty} g_{1,i} D^i + \sum_{i=0}^{\infty} g_{2,i} D^i = \sum_{i=0}^{\infty} (g_{1,i} + g_{2,i}) D^i,$$

we have, for all $i \geq 0$,

$$g_{i+p} = g_{i+p_1 p_2} = g_{1,i+p_2 p_1} + g_{2,i+p_1 p_2}.$$

$G_1(D)$ has period p_1, then we have $g_{1,i+p_2 p_1} = g_{1,i}$. Similarly, $G_2(D)$ has period p_2, then $g_{2,i+p_1 p_2} = g_{2,i}$. Therefore,

$$g_{i+p} = g_{1,i} + g_{2,i} = g_i,$$

i.e., $G(D)$ has period $p = p_1 p_2$.

Note that a stronger statement is true: If p_1 and p_2 are the *least* periods of $G_1(D)$ and $G_2(D)$, respectively, then the *least* period of $G(D)$ is $\mathrm{lcm}(p_1, p_2)$.

Exercise 3.2
From class, we have,

$$g_0(D) \quad = \quad \sum_{i=1}^{r} \sum_{j=-i}^{-1} c_i s_j D^{i+j},$$

where the initial state is indicated by the r-uple $S_0 = (s_{-r}, s_{-r+1}, \cdots, s_{-1})$. We can write,

$$\begin{aligned} g_0(D) \quad &= \quad \sum_{i=1}^{r} \sum_{l=0}^{i-1} c_i s_{l-i} D^l \\ &= \quad \sum_{l=0}^{r-1} \Big(\sum_{i=1}^{r} c_i s_{l-i} \Big) D^l. \end{aligned}$$

One-to-one correspondance between polynomials and initial states :
Using $g_0(D) = \sum_{l=1}^{r-1} g_{0,l} D^l$ and identifying coefficient by coefficient, we get a square system of $r-1$ equations and $r-1$ unknowns $(s_i)_{-r \leq i \leq -1}$ over the binary field. It has one and only one solution. Therefore the mapping between polynomials $g_0(D)$ and initial states S_0 is bijective.

Conclusion :
Since the polynomial $c(D)$ is primitive, it generates the maximum length LFSR. The period is $2^r - 1$ for all non-zero initial state. Two different non-zero polynomials $g_1(D)$ and $g_2(D)$ correspond to two different non-zero initial states. Let's call them S_1 and S_2, respectively. Since all the cycle composed by the non-zero state will be described by the successive states of the LFSR, the two sequences $G_1(D)$ and $G_2(D)$ will be simply delayed versions of each other.

Linearity of the correspondance between polynomials and initial states :
The linearity over the binary field of the mapping between polynomials $g_0(D)$ and initials states S_0 comes also clearly from the previous equation. (Take simply two polynomials $g_0(D)$ and $\tilde{g}_0(D)$ associated to states S_0 and $\tilde{S}_0$, and see that their sum is assosiated to the state $S_0 + \tilde{S}_0 = (s_{-r}, s_{-r+1}, \cdots, s_{-1}) + (\tilde{s}_{-r}, \tilde{s}_{-r+1}, \cdots, \tilde{s}_{-1})$.

Conclusion :
The sequences $G_1(D)$, $G_2(D)$ and $G_1(D) + G_2(D) = \frac{g_1(D)+g_2(D)}{c(D)}$ will also be delayed versions of each other since the sequence $G_1(D) + G_2(D)$ simply starts from an initial state which is $S_1 + S_2$. (Once more all the cycle composed by the non-zero initial states is visited.)

Exercise 3.3
Consider two *non-trivially shifted* versions of a *non-zero* output sequence which agree at a given bit position. Since the MLSR has maximum length, they correspond to two different, non-zero initial states which agree at a given position j. The sum of those two output sequences is a non-zero sequence which is a shifted version of one of them (See Ex. 3.2, the memory r MLSR has maximum length : starting in state S_{sum}, it goes through all its non-zero zero states during a run cycle.). Moreover, this sum corresponds to an initial state S_{sum} which has a 0 at a given position j .(If the two output sequences agree by 1, we have $1 + 1 = 0 = s_{sum,\ j}$ at the given position j in the initial state. If they agree by 0, we have $0 + 0 = 0$ in the initial state.)
We have $2^{r-1} - 1$ non-zero states with 0 at the given position j. We have $2^r - 1$ non-zero states. Therefore, we can clearly state that the two versions will agree at a given bit position with probability $\frac{2^{r-1}-1}{2^r-1}$

Exercise 3.5
Chernov bound :

We have, for all $s > 0$,

$$\Pr\{\sum_{i=1}^{n} X_i > \alpha\} = \Pr\{e^{s\sum_{i=1}^{n} X_i} > e^{s\alpha}\},$$

so that, using the Markov inequality,

$$\Pr\{\sum_{i=1}^{n} X_i > \alpha\} \leq \frac{\mathbb{E}[e^{s\sum_{i=1}^{n} X_i}]}{e^{s\alpha}}.$$

This is true for all $s > 0$, therefore

$$\Pr\{\sum_{i=1}^{n} X_i > \alpha\} \leq \min_{s>0}\Big(\frac{\mathbb{E}[e^{s\sum_{i=1}^{n} X_i}]}{e^{s\alpha}}\Big).$$

Since $X_1, \cdots, X_n$ are i.i.d. random variables, the random variables $e^{sX_1}, \cdots, e^{sX_n}$ are also i.i.d., we get finally,

$$\Pr\{\sum_{i=1}^{n} X_i > \alpha\} \leq \min_{s>0}\Big(\mathbb{E}[e^{sX_1}]^n e^{-s\alpha}\Big).$$

Exponential bound for binary variables :
Consider now a binary random variable taking values in $\{-1, +1\}$ with equal probability. The Chernov bound $\Pr\{\sum_{i=1}^{n} X_i > \alpha\} \leq \min_{s>0}\Big((\cosh(s))^n e^{-s\alpha}\Big)$ can be written $\Pr\{\sum_{i=1}^{n} X_i > \alpha\} \leq \min_{s>0}\Big(f(s)^n\Big)$ using $\gamma = \frac{\alpha}{n}$ (Notice that α is a function of n) and the function $f(s) = (\cosh(s))e^{-s\gamma}$. Computing $f'(s) = \Big(\sinh(s) - \gamma\cosh(s)\Big)e^{-\gamma s}$, we can find the minimum of the function f. We get the upperbound,

$$\begin{aligned}\Pr\{\sum_{i=1}^{n} X_i > \alpha(n)\} &\leq \Big[\cosh(\text{atanh}\gamma)\exp(-\gamma\,\text{atanh}\gamma)\Big]^n,\\ &\leq \Big[\exp\Big(\frac{(\text{atanh}\gamma)^2}{2} - \gamma\,\text{atanh}\gamma\Big)\Big]^n,\end{aligned}$$

using the inequality $\cosh(x) \leq e^{\frac{x^2}{2}}$ which may be obtained by comparing termwise the Taylor series. For all the γ such that $0 \leq \gamma \leq \frac{1}{2}$, we clearly got an exponential bound which is of interest and tends to 0 when n goes to infinity. I.e., for all $\gamma \leq 0.5$, we can find a $k_\gamma < 1$ such that,

$$\Pr\{\sum_{i=1}^{n} X_i > \gamma n\} \leq k_\gamma^n.$$

Remark :
From the Chernov bound $\Pr\{\sum_{i=1}^{n} X_i > \alpha\} \leq \min_{s>0}\Big((\cosh(s))^n e^{-s\alpha}\Big)$, using $\cosh(x) \leq e^{x^2/2}$, we may also get the useful non-exponential bound,

$$\Pr\{\sum_{i=1}^{n} X_i > \alpha\} \leq \min_{s<0}\Big(\exp(\frac{ns^2}{2} - s\alpha)\Big) = e^{\frac{\alpha^2}{2n}},$$

which is of interest for values of the type $\alpha = \hat{\gamma}\sqrt{n}$.

Bound for i.i.d. Gaussian random variables :
Consider Gaussian random variables with i.i.d. distributions $\mathcal{N}(0,\sigma^2)$. We have,

$$\begin{aligned}
\mathbb{E}[\mathrm{e}^{sX_1}] &= \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\sigma} \mathrm{e}^{-\frac{x^2}{2\sigma^2}} \mathrm{e}^{sx} \mathrm{d}x \\
&= \mathrm{e}^{\frac{s^2\sigma^2}{2}} \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\sigma} \mathrm{e}^{-\frac{x^2-2\sigma^2 sx+(s\sigma^2)^2}{2\sigma^2}} \mathrm{d}x \\
&= \mathrm{e}^{\frac{s^2\sigma^2}{2}} \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\sigma} \mathrm{e}^{-\frac{(x-s\sigma^2)^2}{2\sigma^2}} \mathrm{d}x \\
&= \mathrm{e}^{\frac{s^2\sigma^2}{2}} \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\sigma} \mathrm{e}^{-\frac{(y)^2}{2\sigma^2}} \mathrm{d}y \\
&= \mathrm{e}^{\frac{s^2\sigma^2}{2}}.
\end{aligned}$$

Using the Chernov bound,we get

$$\Pr\{\sum_{i=1}^{n} X_i > \alpha\} \leq \min_{s>0} \left(\mathrm{e}^{\frac{ns^2\sigma^2}{2} - s\alpha}\right) = \min_{s>0} \left(\mathrm{e}^{s(\frac{ns\sigma^2}{2} - \alpha)}\right) = \exp\Big(-\frac{\alpha^2}{2n\sigma^2}\Big),$$

which is of interest for $\alpha = \gamma'\sqrt{n}$.

Remark :
This result can be compared with the Q-function which is an upperbound for the Gaussian case. Indeed, we know that the random variable $\sum_{i=1}^{n} X_i$ which is a sum of Gaussian random variables with i.i.d. distributions $\mathcal{N}(0,\sigma^2)$ is a Gaussian random variable with distribution $\mathcal{N}(0,n\sigma^2)$. Therefore we have directly the upper bound,

$$\Pr\{\sum_{i=1}^{n} X_i > \alpha\} \leq \int_{\alpha}^{\infty} \frac{1}{\sqrt{2\pi}\sqrt{n}\sigma} \mathrm{e}^{-\frac{x^2}{2n\sigma^2}} \mathrm{d}x = Q(\frac{\alpha}{\sqrt{n}\sigma}),$$

so that, using the upperbound of Ex. 1.5, we get

$$\Pr\{\sum_{i=1}^{n} X_i > \alpha\} \leq \exp\Big(-\frac{\alpha^2}{2n\sigma^2}\Big),$$

i.e., the same result as obtained using the Chernov bound technique.

Exercise 3.6
Assume we allow *each* user to scale his input signal by a factor strictly α, i.e., the baseband signal is given by

$$\alpha\sqrt{E_c}\sum_n x_n s_n h(t - nT_c).$$

In our original anaylis E_c was an arbitrary constant. So for the new system we just have to replace E_c with $\alpha^2 E_c$. But assuming the background noise is neglible, the performance of the system is not changed: this is true since although the energy of the signal of interest is changed by a factor α^2 the interference caused by other users is also boosted by the same factor. In a well-designed system one should therefore probably operate at the smallest power for which the background noise is smaller than the noise caused by the other user's interference.

Exercise 3.7
The steps of the analysis stay the same except that we get the noise variance,

$$\sigma^2 = \sum_{j=1}^{k} \frac{E(j)}{NT_c} \int_{-\infty}^{\infty} |H(f)|^4 \mathrm{d}f = (k-1)\frac{E}{N}.$$

But note that in this case the noise is *real-valued*! Therefore the bit error probability is changed to $Q(\sqrt{\frac{E}{N_0}})$ (instead of $Q(\sqrt{\frac{2E}{N_0}})$ previously). Since $\frac{N}{k-1}$ corresponds to $\frac{E}{N_0}$, the number of supportable users is half what it was for the original system.

Remark :
Two independent real-valued random variables A and B define a complex-valued random variable Y as

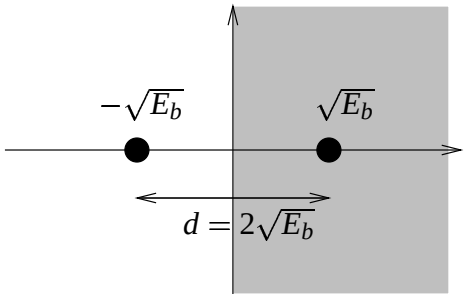
$$Y = A + jB,$$

so that we have $\sigma_Y^2 = \sigma_A^2 + \sigma_B^2$ and $\sigma_Y^2 = 2\sigma_A^2 = 2\sigma_B^2$ if A and B are i.i.d..

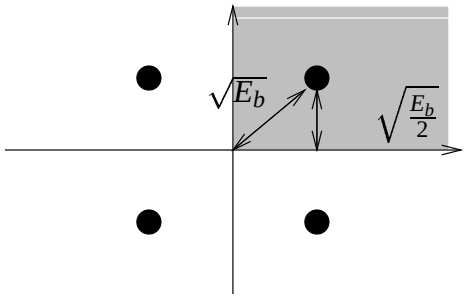
Exercise 3.8
Once more the analysis stays the same except that the constellation in Fig. 8.1 (i) is replaced by the constellation in Fig. 8.1 (ii). Note that now the minimum distance between points is smaller by a factor $\sqrt{2}$. Therefore we can only support half the number of users as before assuming that again background noise is negligible. We see that the overall bit rate (sum of all individual bit rates) is again unchanged since now we have half the number of users transmitting at twice the individual bit rate.

Exercise 3.9

1. The total noise power is just the integral $\mathcal{S}_Z(f)$ over $\mathbb{R}$, which is equal to $2\frac{N_0 W}{2B}B = N_0 W$.



(i)



(ii)

Figure 8.1: BPSK and QPSK Constellation and Decision Region.

2. The signal portion is equal to

$$\begin{aligned}\int x(t)g_i^*(t-iT)dt &= \int \sqrt{E}\sum_j u_j g_j(t-jT)g_i^*(t-iT)dt \\ &= \sqrt{E}\sum_i u_i \int g_i(t-jT)g_i^*(t-iT)dt \\ &= \sqrt{E}u_i,\end{aligned}$$

where we have used the fact that the functions $g_i(t-iT)$ are orthonormal.

3.

$$\begin{aligned}z_i &= \int Z(t)g_i^*(t-iT)dt \\ &= \int Z(t)\frac{1}{\sqrt{N}}\sum_{n=0}^{N-1} s_{iN+n}^* h(t-nT_c)dt \\ &= \frac{1}{\sqrt{N}}\sum_{n=0}^{N-1} s_{iN+n}^* \int Z(t)h(t-nT_c)dt.\end{aligned}$$

4. Note that the random variables W_{iN+n} is the result of passing $Z(t)$ through a linear time invariant filter with impulse response $h(-t)$ and sampling at time $iT+nT_c$. From this, and the properties of $Z(t)$ it follows that W_{iN+n} is a complex-valued circularly symmetric Gaussian random variable with zero mean. The filtered process has a power spectral density which has the same shape as the power spectral density of $Z(t)$ but its magnitude is smaller by a factor $\frac{1}{W}$ since $|H(f)| = \frac{1}{\sqrt{w}}$ over the frequency region of interest. Therefore the variance of W_{iN+n}, which is equal to the integral of the power spectral density of the filtered process is equal to $2\frac{N_0}{2B}B = N_0$.

5. z_i is the finite sum of complex-valued circularly symmetric Gaussian random variable with zero mean and, therefore, is a complex-valued circularly symmetric Gaussian random variable with zero mean itself. Finally, we have

$$\begin{aligned}\mathbb{E}[z_i z_j^*] &= \mathbb{E}\left[\left(\frac{1}{\sqrt{N}}\sum_{n=0}^{N-1} s_{iN+n}^* W_{iN+n}\right)\left(\frac{1}{\sqrt{N}}\sum_{m=0}^{N-1} s_{jN+m} W_{jN+m}^*\right)\right] \\ &= \frac{1}{N}\sum_{n=0}^{N-1}\sum_{m=0}^{N-1}\mathbb{E}[s_{iN+n}^* s_{jN+m}]\mathbb{E}[W_{iN+n}W_{jN+m}^*] \\ &= \frac{1}{N}\sum_{n=0}^{N-1}\sum_{m=0}^{N-1}\delta_{i-j}\delta_{n-m}\mathbb{E}[W_{iN+n}W_{jN+m}^*] \\ &= \delta_{i-j}\frac{1}{N}\sum_{n=0}^{N-1}\mathbb{E}[W_{iN+n}W_{iN+n}^*] \\ &= N_0\delta_{i-j}.\end{aligned}$$

Bibliography

[1] A. Shokrollahi, “New sequences of linear time erasure codes approaching the channel capacity,” in *Proceedings of AAECC-13, Lecture Notes in Computer Science 1719*, pp. 65–76, 1999.

A

LINEAR PREDICTION

Assume we have two (correlated) zero mean WSS stochastic processes x_n and y_n and we would like to *predict* x_n based upon the observations $\{y_k\}_{k\in n-\mathcal{K}}$, where $\mathcal{K}$ is some given set of indices. Although our setup encompasses more general scenarios we will in particular be interested in the following two examples.

Example 23. In the first scenario we have $y_n = x_n$ and $\mathcal{K} = \{k : k > 0\}$. In words, we want to predict x_n based upon the observation of its *past* samples.

Example 24. In the second scenario we have $y_n \neq x_n$ but now we allow $\mathcal{K} = \mathbb{Z}$.

We are interested in *linear* prediction, i.e., our prediction has the form

$$\sum_{k\in\mathcal{K}} h_k y_{n-k}.$$

Define the *error sequence* e_n as

$$e_n := x_n - \sum_{k\in\mathcal{K}} h_k y_{n-k}. \tag{A.1}$$

Clearly, we will be interested in keeping the error "as small as possible." More precisely, we choose as a criterion to minimize the *mean-squared error*, i.e., to minimize $\mathbb{E}[|e_n|^2]$. Therefore this criterion is often abbreviated as MMSE. How should we choose the filter coefficients in order to minimize $\mathbb{E}[|e_n|^2]$? Let $\mathcal{R}_x(k) := \mathbb{E}[x_n x_{n-k}^*]$, $\mathcal{R}_y(k) := \mathbb{E}[y_n y_{n-k}^*]$ and $\mathcal{R}_{x,y}(k) := \mathbb{E}[x_n y_{n-k}^*] = \mathcal{R}_{x,y}^*(-k)$.

We claim that we should choose the filter coefficients in such a way that

$$\mathbb{E}[e_n y_{n-k}^*] = 0, \;\; k \in \mathcal{K}. \tag{A.2}$$

This is called the *orthogonality principle*. The intuition behind this choice is that if there were some remaining correlation then we could use this correlation to

perform a better prediction! To see this claim, let e'_n denote the error sequence associated to any linear predictor based upon the set of observations $\{y_k\}_{k\in n-\mathcal{K}}$ and let e_n denote the error sequence associated to the linear predictor derived by the orthogonality principle. Then we have

$$\begin{aligned}
\mathbb{E}[|e'_n|^2] &= \mathbb{E}[|(e'_n - e_n) + e_n|^2] \\
&= \mathbb{E}[|(e'_n - e_n)|^2] + \mathbb{E}[|e_n|^2] + 2\,\mathrm{Re}\,\mathbb{E}[(e'_n - e_n)e_n^*] \\
&= \mathbb{E}[|(e'_n - e_n)|^2] + \mathbb{E}[|e_n|^2] + \\
&\quad 2\,\mathrm{Re}\left\{\mathbb{E}[((x_n - \sum_{k\in\mathcal{K}} h'_k y_{n-k}) - (x_n - \sum_{k\in\mathcal{K}} h_k y_{n-k}))e_n^*]\right\} \\
&= \mathbb{E}[|(e'_n - e_n)|^2] + \mathbb{E}[|e_n|^2] + 2\,\mathrm{Re}\left\{\mathbb{E}[(\sum_{k\in\mathcal{K}} (h_k - h'_k)y_{n-k})e_n^*]\right\} \\
&= \mathbb{E}[|(e'_n - e_n)|^2] + \mathbb{E}[|e_n|^2] + 2\,\mathrm{Re}\left\{\sum_{k\in\mathcal{K}} (h_k - h'_k)\mathbb{E}[e_n y_{n-k}^*]^*\right\} \\
&= \mathbb{E}[|(e'_n - e_n)|^2] + \mathbb{E}[|e_n|^2] \\
&\geq \mathbb{E}[|e_n|^2].
\end{aligned}$$

In order to find the coefficients of the filter note that for $k \in \mathcal{K}$

$$\begin{aligned}
\mathbb{E}[e_n y_{n-k}^*] &= \mathbb{E}[(x_n - \sum_{m\in\mathcal{K}} h_m y_{n-m})y_{n-k}^*] \\
&= \mathbb{E}[x_n y_{n-k}^*] - \sum_{m\in\mathcal{K}} h_m \mathbb{E}[y_{n-m}y_{n-k}^*] \\
&= \mathcal{R}_{x,y}(k) - \sum_{m\in\mathcal{K}} h_m \mathcal{R}_y(k-m).
\end{aligned}$$

Therefore, we have the defining equations

$$\mathcal{R}_{x,y}(k) = \sum_{m\in\mathcal{K}} h_m \mathcal{R}_y(k-m), \;\; k \in \mathcal{K}. \tag{A.3}$$

Example 25. Consider first the case $x_n = y_n$ and $\mathcal{K} = \mathbb{Z}$. Clearly, this is a trivial example since in this case we are allowed to look at the very same sequence (in its entirety) which we want to predict and therefore it is obvious that we can achieve perfect prediction! Nevertheless, proceeding formally, the defining equations in this case specialize to

$$\mathcal{R}_x(k) = \sum_{m=-\infty}^{\infty} h_m \mathcal{R}_x(k-m), \;\; k \in \mathbb{Z}.$$

Taking the z-transform this is equivalent to

$$\mathcal{S}_x(z) = \mathcal{S}_x(z)H(z).$$

The solution is obviously $H(z) = 1$, or $h_n = \delta(n)$, as expected.

Example 26. Consider now the case $x_n = y_n$ and $\mathcal{K} = \{k : k > 0\}$. Let's first derive the optimal filter in an alternative way. We claim that in this case the process e_k is itself uncorrelated. To see this note that

$$\begin{aligned}
\mathcal{R}_e(k) &= \mathbb{E}[e_n e_{n-k}^*] \\
&= \mathbb{E}[e_n (x_{n-k} - \sum_{m=1}^{\infty} h_m x_{n-k-m})^*] \\
&= \mathbb{E}[e_n x_{n-k}^*] - \sum_{m=1}^{\infty} h_m^* \mathbb{E}[e_n x_{n-k-m}^*] \\
&= 0.
\end{aligned}$$

Note that from (A.1), $E(z) = F(z)X(z)$, where $F(z) = 1 - H(z)$ and where $H(z)$ is strictly causal. Since by the above derivation e_n is white, we see that f_n is the monic and causal whitening filter. From $H(z) = 1 - F(z)$, $H(z)$ follows trivially.

Alternatively, we can start from the defining equations given in (A.3) which specialize for the present case to

$$\sum_{m \geq 0} f_m \mathcal{R}_x(k-m) = 0, \;\; k \geq 1, \tag{A.4}$$

where we defined the new filter f_n related to h_n by

$$f_n := \begin{cases} 0, & n < 0, \\ 1, & n = 0, \\ -h_n, & n \geq 1. \end{cases}$$

It is not immediate obvious that this implies that f_n should be the monic and causal whitening filter. You will show in Exercise 2.9 that this is indeed true.

Example 27. As a final example consider the case $x \neq y$ and $\mathcal{K} = \mathbb{Z}$. Since $\mathcal{K}$ equals $\mathbb{Z}$ we can take the z-transform of the defining equations and derive at

$$\mathcal{S}_{x,y}(z) = H(z)\mathcal{S}_y(z).$$

Therefore, the optimal filter in this case is given by

$$H(z) = \frac{\mathcal{S}_{x,y}(z)}{\mathcal{S}_y(z)}. \tag{A.5}$$

B

SPECTRAL FACTORIZATION

Let $\mathcal{R}(k)$ be a correlation function and assume that the associated spectrum $\mathcal{S}(z)$ is *rational*. Observe that $\mathcal{R}(k) = \mathcal{R}^*(-k)$, which implies that $\mathcal{S}(z) = \mathcal{S}^*(1/z^*)$. We conclude that the roots (zeros) of $\mathcal{S}(z)$ have the symmetry that if ρ is a pole (zero) then so is $1/\rho^*$. We say that ρ and $1/\rho^*$ are *conjugate-symmetric*. This symmetry relationship is shown in Fig. B.1. Note that $\mathcal{S}(z)$ can not have poles on

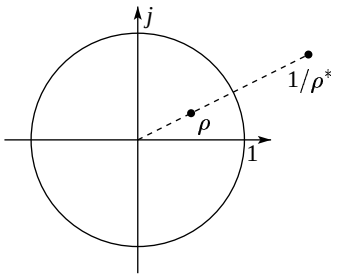


Figure B.1: A conjugate-symmetric pair ρ and $1/\rho^*$.

the unit circle since it is easy to check that otherwise

$$\mathcal{R}(0) := \int_{-\frac{1}{2}}^{\frac{1}{2}} \mathcal{S}(e^{2\pi jf}) df = \infty.$$

Zeros, on the other hand can be located on the unit circle but one can show that such zeros have to appear in pairs also.

From these observations it follows that $\mathcal{S}(z)$ has the form

$$\mathcal{S}(z) = A^2 \frac{\prod_{k=1}^{M}(1 - c_k z^{-1})(1 - c_k^* z)}{\prod_{k=1}^{N}(1 - d_k z^{-1})(1 - d_k^* z)},$$

where $|c_k| \leq 1$, $|d_k| < 1$ and where $A \in \mathbb{R}$. Therefore $\mathcal{S}(z)$ can be factored as

$$\begin{aligned} \mathcal{S}(z) &= \left[A\frac{\prod_{k=1}^{M}(1-c_k z^{-1})}{\prod_{k=1}^{N}(1-d_k z^{-1})}\right]\left[A\frac{\prod_{k=1}^{M}(1-c_k^* z)}{\prod_{k=1}^{N}(1-d_k^* z)}\right] \\ &= \mathcal{S}^+(z)\mathcal{S}^-(z) = \mathcal{S}^+(z)\left(\mathcal{S}^+(1/z^*)\right)^*, \end{aligned} \tag{B.1}$$

where $\mathcal{S}^+(z)$ is *causal* and, $\mathcal{S}^-(z)$ is anticausal.

Now,

$$\begin{aligned} &\int_{-\frac{1}{2}}^{\frac{1}{2}} \ln\left[\mathcal{S}(e^{2\pi jf})\right] df = \ln A^2 + \\ &\sum_{k=1}^{M}\int_{-\frac{1}{2}}^{\frac{1}{2}} \ln\left(1-c_k e^{-2\pi jf}\right) df + \sum_{k=1}^{M}\int_{-\frac{1}{2}}^{\frac{1}{2}} \ln\left(1-c_k^* e^{2\pi jf}\right) df - \\ &\sum_{k=1}^{N}\int_{-\frac{1}{2}}^{\frac{1}{2}} \ln\left(1-d_k e^{-2\pi jf}\right) df - \sum_{k=1}^{N}\int_{-\frac{1}{2}}^{\frac{1}{2}} \ln\left(1-d_k^* e^{2\pi jf}\right) df \\ = \ &\ln A^2. \end{aligned}$$

To see the last step note the following. If $a \in \mathbb{C}$, $|a| \leq 1$, then

$$\begin{aligned} &\int_{-\frac{1}{2}}^{\frac{1}{2}} \ln\left((1-ae^{-2\pi jf})(1-a^*e^{2\pi jf})\right) df \\ = \ &\int_{-\frac{1}{2}}^{\frac{1}{2}} \ln\left(1+|a|^2-\left(ae^{-2\pi jf}+a^*e^{2\pi jf}\right)\right) df \\ = \ &\int_{-\frac{1}{2}}^{\frac{1}{2}} \ln\left(1+|a|^2-2|a|\left(e^{-2\pi jf+\phi_a}+e^{2\pi jf-\phi_a}\right)\right) df \\ = \ &\int_{-\frac{1}{2}}^{\frac{1}{2}} \ln\left(1+|a|^2-2|a|\cos(2\pi f-\phi_a)\right) df \\ = \ &\int_{-\frac{1}{2}}^{\frac{1}{2}} \ln\left(1+|a|^2-2|a|\cos(2\pi f)\right) df \\ = \ &0, \end{aligned}$$

where the last integral can be found in standard integral tables. Alternatively, if $|a| < 1$ then we can argue that the function $\frac{\ln(1-az)}{z}$ is analytic for $|z| \leq 1$. Therefore, by Chauchy's formula

$$\int_{-\frac{1}{2}}^{\frac{1}{2}} \ln(1-ae^{\pm 2\pi jf})df = \frac{1}{2\pi j}\int_{|z|=1}\frac{\ln(1-az)}{z}dz = 0.$$

It follows that

$$A^2 = \exp\left\{\int_{-\frac{1}{2}}^{\frac{1}{2}} \ln\left[\mathcal{S}(e^{2\pi jf})\right] df\right\}. \tag{B.2}$$

Observe from (B.1) that $\mathcal{S}^{\pm}(z)$ is the product of the constant A with a monic causal/anticausal filter. Therefore,

$$\frac{\mathcal{S}^{\pm}(z)}{\sqrt{\exp\left\{\int_{-\frac{1}{2}}^{\frac{1}{2}} \ln[\mathcal{S}(e^{2\pi jf})]\,df\right\}}} \tag{B.3}$$

is a monic causal filter. This observation, which we derived for rational spectra, plays an important role in the analysis of equalizers.

So far we have assumed the spectrum is rational. But the spectral factorization as well as formula (B.2) are not restricted to the rational case. We will now give a more general formulation.

Lemma 10. Let $\mathcal{R}(k)$ be a correlation function and $\mathcal{S}(z)$ the associated spectrum. If $\mathcal{S}(z)$ satisfies the *Paley-Wiener* condition

$$\int_{-\frac{1}{2}}^{\frac{1}{2}} \ln \mathcal{S}(e^{2\pi jf})df > -\infty,$$

then $\mathcal{S}(e^{2\pi jf})$ can be written as the product of two functions $\mathcal{S}^{+}(e^{2\pi jf})$ and $\mathcal{S}^{-}(e^{2\pi jf})$ such that

1. $|\mathcal{S}^{+}(e^{2\pi jf})|^2 = |\mathcal{S}^{-}(e^{2\pi jf})|^2 = \mathcal{S}(e^{2\pi jf})$.
2. $\mathcal{S}^{+}(e^{2\pi jf})$ is causal and $\mathcal{S}^{-}(e^{2\pi jf})$ is anticausal.
3. $\frac{1}{\mathcal{S}^{+}(e^{2\pi jf})}$ is causal and $\frac{1}{\mathcal{S}^{-}(e^{2\pi jf})}$ is anticausal.

Proof. Note that $\ln \mathcal{S}(e^{2\pi jf})$ is a periodic function. Write it as

$$\ln \mathcal{S}(e^{2\pi jf}) = \sum_n c_n e^{-2\pi jfn}.$$

Therefore

$$\begin{aligned}
\mathcal{S}(e^{2\pi jf}) &= e^{\sum_n c_n e^{2\pi jfn}} \\
&= e^{c_0/2+\sum_{n\geq 1} c_n e^{-2\pi jfn}} e^{c_0/2+\sum_{n\leq -1} c_n e^{-2\pi jfn}} \\
&=: \mathcal{S}^{+}(e^{2\pi jf})\mathcal{S}^{-}(e^{-2\pi jf}).
\end{aligned}$$

It remains to show that $\mathcal{S}^{+}(e^{2\pi jf})$ and $\mathcal{S}^{-}(e^{2\pi jf})$ realy have the claimed properties.

Since $\mathcal{S}(e^{-2\pi jf})$ is a power spectral density it is real and even. It follows that $\ln \mathcal{S}^{+}(e^{2\pi jf})$ is real and even. This implies that $c_{-n} = c_n = c_n^*$. Therefore

$$\begin{aligned}
\mathcal{S}^{-}(e^{2\pi jf}) &= e^{c_0/2+\sum_{n\leq -1} c_n e^{-2\pi jfn}} \\
&= e^{c_0/2+\sum_{n\geq 1} c_n e^{2\pi jfn}} \\
&= e^{c_0/2+\sum_{n\geq 1} c_n e^{-2\pi jfn^*i}} \\
&= \left(\mathcal{S}^{+}(e^{2\pi jf})\right)^*.
\end{aligned}$$

It follows that $|\mathcal{S}^+(e^{2\pi jf})|^2 = |\mathcal{S}^-(e^{2\pi jf})|^2$. It remains to show that $\mathcal{S}^+(e^{2\pi jf})$ is causal. Using the Taylor series expansion of e^x, $e^x = \sum_{k\geq 0} \frac{1}{k!}x^k$ we obtain

$$\mathcal{S}^+(e^{2\pi jf}) = e^{c_0/2}\left[\sum_{k=0}^{\infty}\frac{1}{k!}\left(\sum_{n=1}^{\infty}c_n e^{-2\pi jfn}\right)^k\right].$$

This shows that $\mathcal{S}^+(e^{2\pi jf})$ is causal. □

Index