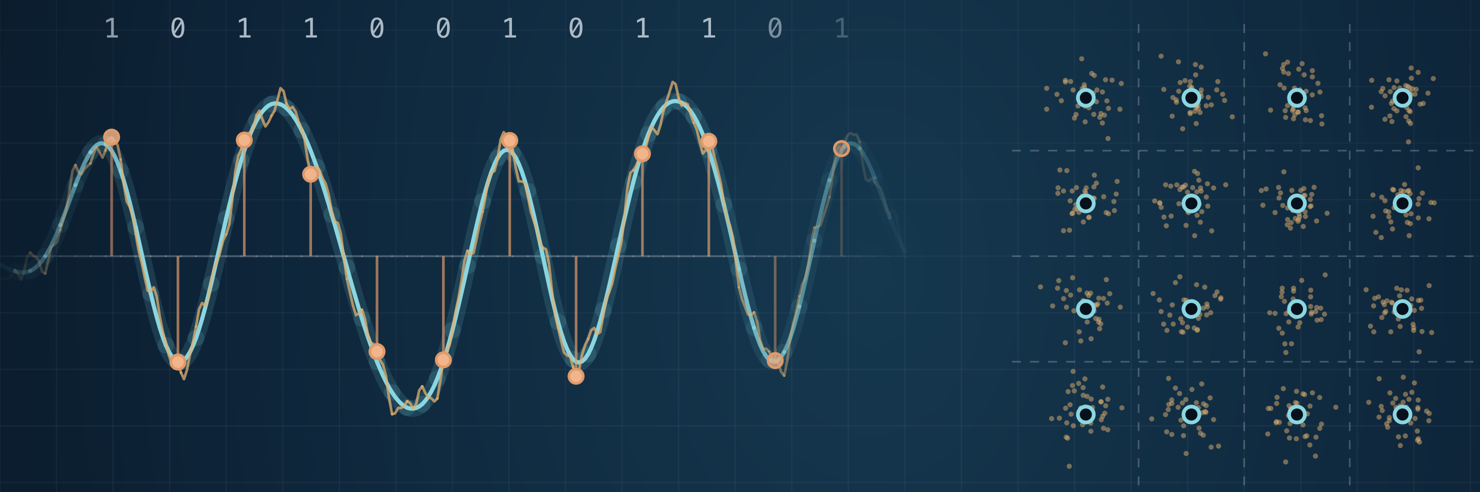
SAMPLING · DETECTION · MODULATION · CODING

Digital Communications

Lecture Notes



Hüseyin Uğur Yıldız

IEEE Senior Member
Associate Professor of Electrical and Electronics Engineering
TED University, Ankara, Türkiye

CHAPTERS 1–6 · APPENDIX A

Contents

Course conventions

These notes are written for undergraduates. They assume Fourier analysis, probability and random processes. Read the plain explanation before the mathematics. Worked examples use five headings: Given, Find, Method, Solution, and Check.

Noise is white and Gaussian with **two-sided** power spectral density $N_0/2$ watts per hertz. The Gaussian tail is $Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-t^2/2}\,dt = \frac{1}{2}\,\mathrm{erfc}(x/\sqrt{2})$. Energy and power use $R = 1\ \Omega$.

A **PS** marker points to the related section of Proakis and Salehi, *Fundamentals of Communication Systems*, second edition. The textbook chapter numbers differ from the course chapter numbers.

List of Figures

CHAPTER 1 · THE TRANSITION FROM ANALOG TO DIGITAL

CHAPTER 4 · THE OPTIMAL RECEIVER IN ADDITIVE WHITE GAUSSIAN NOISE

CHAPTER 5 · DIGITAL MODULATION METHODS

CHAPTER 6 · AN INTRODUCTION TO INFORMATION THEORY

List of Tables

CHAPTER

1

The transition from analog to digital

A continuous waveform becomes a bit stream in three stages: sampling, quantization and encoding.

Two results carry through the chapter. Sampling can be undone. At $f_s \geq 2W$ the samples of a signal bandlimited to W determine it exactly. Quantization cannot be undone. Each extra bit divides the error power by four, which is about 6 dB of signal-to-quantization-noise ratio.

1.1 The sampling theorem

Fourier transform review

This chapter writes the Fourier transform in the frequency variable f, in hertz.

THE TRANSFORM IN f

$$X(f) = \int_{-\infty}^{\infty} x(t)\, e^{-j2\pi f t}\, dt$$
$$x(t) = \int_{-\infty}^{\infty} X(f)\, e^{j2\pi f t}\, df$$

Put $\omega = 2\pi f$. Then $d\omega = 2\pi\, df$, so the factor $1/2\pi$ of the inverse transform in ω disappears.

Five properties are used below. A shift in time multiplies the transform by a phase. A multiplication by a complex exponential shifts the transform. Convolution and multiplication trade places.

PROPERTIES

$$\begin{aligned}
x(t - t_0) &\leftrightarrow X(f)\, e^{-j2\pi f t_0} \\
x(t)\, e^{j2\pi f_0 t} &\leftrightarrow X(f - f_0) \\
x(t) * y(t) &\leftrightarrow X(f)\, Y(f) \\
x(t)\, y(t) &\leftrightarrow X(f) * Y(f) \\
\int_{-\infty}^{\infty} |x(t)|^2\, dt &= \int_{-\infty}^{\infty} |X(f)|^2\, df
\end{aligned}$$

The last line is Parseval's theorem: the energy is the same in both domains.

The pairs below are the ones the chapter uses. Here $\Pi(t/T)$ is 1 for $|t| < T/2$ and 0 elsewhere. The course uses $\mathrm{sinc}(x) = \sin(\pi x)/(\pi x)$ and $f_s = 1/T_s$.

PAIRS

$$
\begin{aligned}
\delta(t) &\leftrightarrow 1 \\
1 &\leftrightarrow \delta(f) \\
e^{j2\pi f_0 t} &\leftrightarrow \delta(f - f_0) \\
\cos(2\pi f_0 t) &\leftrightarrow \tfrac{1}{2}\delta(f - f_0) + \tfrac{1}{2}\delta(f + f_0) \\
\Pi(t/T) &\leftrightarrow T\,\mathrm{sinc}(fT) \\
2W\,\mathrm{sinc}(2Wt) &\leftrightarrow \Pi\left(\frac{f}{2W}\right) \\
\sum_n \delta(t - nT_s) &\leftrightarrow \frac{1}{T_s}\sum_n \delta(f - nf_s)
\end{aligned}
$$

COMMON ERROR

Use $1 \leftrightarrow \delta(f)$, not $1 \leftrightarrow 2\pi\delta(f)$, when the ω pair $1 \leftrightarrow 2\pi\delta(\omega)$ moves to f. The reason is $\delta(2\pi f) = \delta(f)/2\pi$.

Impulse-train sampling

An ideal sampler takes one value of the message $g(t)$ every T_s seconds. The **sampling frequency** is $f_s = 1/T_s$. The sampler is modelled as a multiplication by a train of unit impulses.

IMPULSE TRAIN

$$
p(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT_s), \qquad f_s = \frac{1}{T_s}
$$

The **sampled signal** is the product $g_\delta(t) = g(t)\,p(t)$. Move $g(t)$ inside the sum. Then use the sampling property $g(t)\delta(t - t_0) = g(t_0)\delta(t - t_0)$ on each term.

SAMPLED SIGNAL

$$
\begin{aligned}
g_\delta(t) &= g(t)\,p(t) \\
&= g(t) \sum_{n=-\infty}^{\infty} \delta(t - nT_s) \\
&= \sum_{n=-\infty}^{\infty} g(t)\,\delta(t - nT_s) \\
&= \sum_{n=-\infty}^{\infty} g(nT_s)\,\delta(t - nT_s)
\end{aligned}
$$

Each impulse now carries the sample value at its own time.

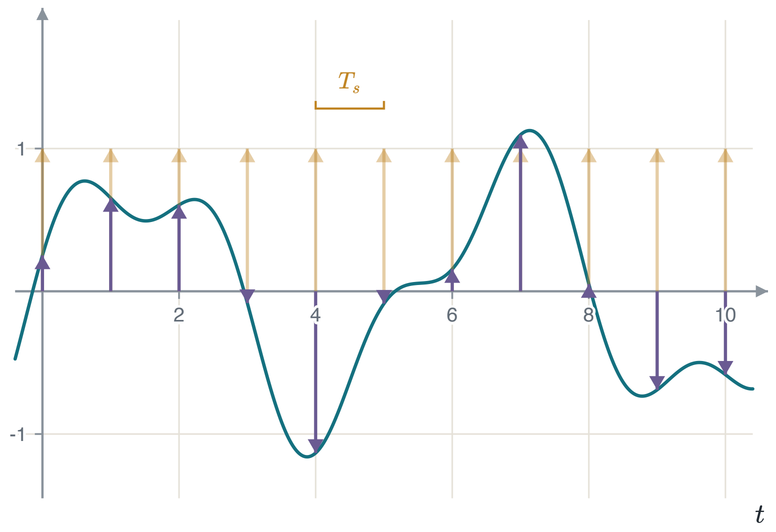


FIGURE 1.1 The message $g(t)$, the impulse train $p(t)$ behind it, and their product $g_\delta(t)$. Each impulse of g_δ has the height of the message at its sampling instant.

For example, a sampler with $T_s = 125\ \mu$s has $f_s = 1/(125 \times 10^{-6}) = 8000$ Hz, or 8 kHz.

The spectrum of a sampled signal

The sampled signal is a product in time. A product in time is a convolution in frequency, so the first step is one line.

> PRODUCT IN TIME
>
> $$G_\delta(f) = G(f) * P(f)$$
>
> In f, no 2π appears in either pair used here: $x(t)\,y(t) \leftrightarrow X(f) * Y(f)$ and $e^{j2\pi f_0 t} \leftrightarrow \delta(f - f_0)$.

The train $p(t)$ is periodic with period T_s, so it has a Fourier series. Over one period it holds only the impulse at $t = 0$. Sifting gives $e^0 = 1$ for every coefficient c_n.

> TRANSFORM OF THE TRAIN
>
> $$c_n = \frac{1}{T_s} \int_{-T_s/2}^{T_s/2} \delta(t)\, e^{-j2\pi n f_s t}\, dt = \frac{1}{T_s}$$
> $$p(t) = \frac{1}{T_s} \sum_{n=-\infty}^{\infty} e^{j2\pi n f_s t}$$
> $$P(f) = \frac{1}{T_s} \sum_{n=-\infty}^{\infty} \delta(f - n f_s)$$
>
> The second pair, with $f_0 = n f_s$, transforms each term of the series.

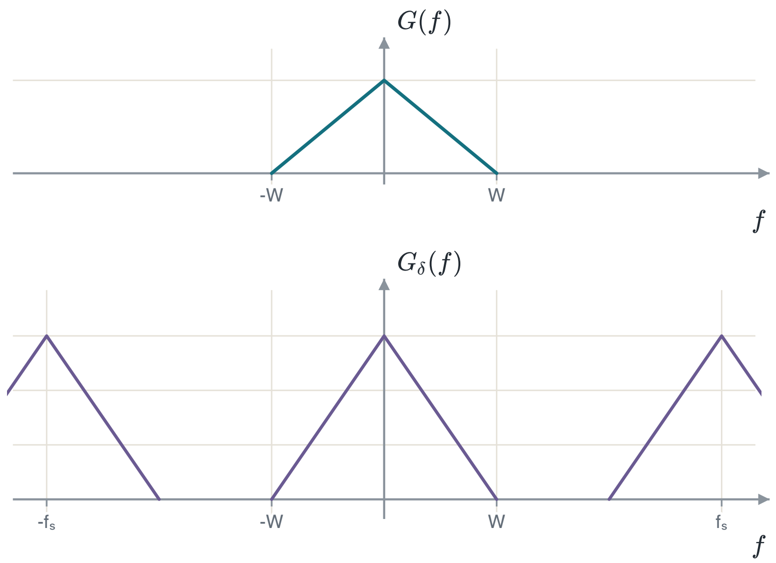


FIGURE 1.2 The message spectrum $G(f)$, and the spectrum after sampling at $f_s = 3W$. A scaled copy of $G(f)$ sits at every multiple of f_s.

For example, let $G(f) = 0$ for $|f| \geq 4$ kHz and $f_s = 10$ kHz. The copy $G(f - f_s)$ occupies $f_s - W < f < f_s + W$. So it begins at $10 - 4 = 6$ kHz.

Spectral replicas

It remains to convolve $G(f)$ with one shifted impulse. Write the convolution integral. The impulse is even, so flip the sign of its argument. Then sift at $\theta = f - nf_s$.

CONVOLUTION WITH ONE IMPULSE

$$
\begin{aligned}
G(f) * \delta(f - nf_s) &= \int_{-\infty}^{\infty} G(\theta)\, \delta(f - nf_s - \theta)\, d\theta \\
&= \int_{-\infty}^{\infty} G(\theta)\, \delta\big(\theta - (f - nf_s)\big)\, d\theta \\
&= G(f - nf_s)
\end{aligned}
$$

Put $P(f)$ into $G(f) * P(f)$ and convolve term by term. Each impulse of $P(f)$ places one copy of G at a multiple of f_s. The factor $1/T_s$ equals f_s.

KEY RESULT: SPECTRUM REPLICAS

$$
\begin{aligned}
G_\delta(f) &= \frac{1}{T_s} \sum_n G(f) * \delta(f - nf_s) \\
&= f_s \sum_{n=-\infty}^{\infty} G(f - nf_s)
\end{aligned}
$$

Sampling copies the spectrum to every multiple of f_s and scales it by f_s.

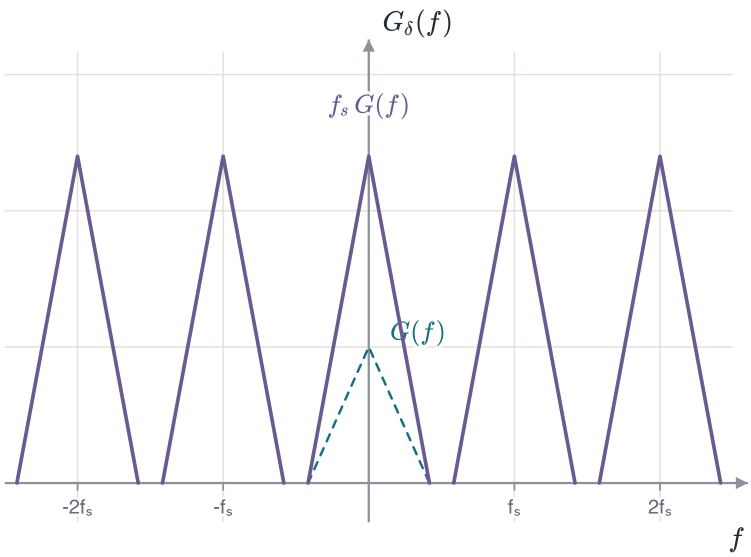


FIGURE 1.3 The message spectrum $G(f)$ (dashed) is scaled by f_s, and one copy is added for each pair of impulses at $\pm n f_s$. Copies are drawn at $\pm f_s$ and $\pm 2f_s$.

For example, let a signal be sampled at $f_s = 8$ kHz. The copy centred at f_s is $G(f - 8\ \mathrm{kHz})$. It equals $G(1\ \mathrm{kHz})$ where $f - 8 = 1$, that is at 9 kHz.

Three sampling rates

The copies have width $2W$ and sit f_s apart. The geometry of the copies separates three cases.

1. $f_s > 2W$: **oversampling**. A gap separates the copies.
2. $f_s = 2W$: **Nyquist sampling**. The copies touch.
3. $f_s < 2W$: **undersampling**. The copies overlap.

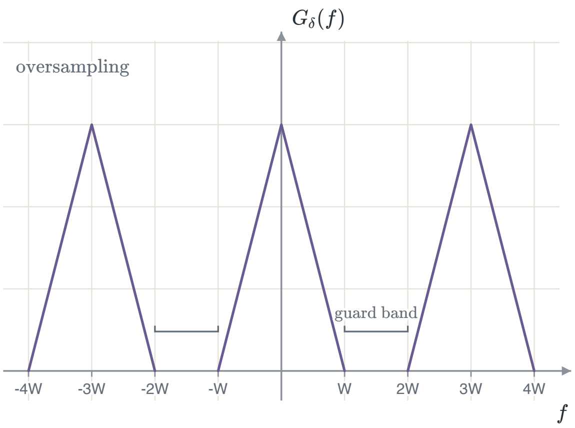


FIGURE 1.4 $f_s = 3W$: a guard band separates the copies.

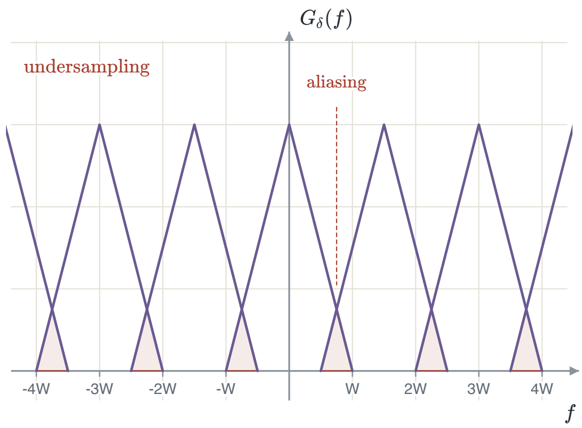


FIGURE 1.5 $f_s = 1.5W$: the copies overlap, and the overlaps (red) are aliasing.

> ⊗ ALIASING
>
> Where two copies overlap, a high message frequency lands on a low one. No filter can separate them, so the samples no longer determine the signal.

⚠ GUARD BAND

A real signal is not strictly bandlimited. Sample at $f_s = 2W + f_g$, where f_g is a **guard band**. Filter out everything above W before the sampler.

For example, let $W = 3$ kHz and $f_s = 5$ kHz. Then $f_s < 2W = 6$ kHz. The first copy starts at $5 - 3 = 2$ kHz, inside the message band, so the copies overlap.

The sampling theorem

The three cases give the condition for exact recovery. The copies must not overlap.

ⓘ SAMPLING THEOREM

If $G(f) = 0$ for $|f| \geq W$ and $f_s \geq 2W$, the samples $g(nT_s)$ determine $g(t)$ exactly.

The copy at f_s starts at $f_s - W$. That start must not lie below the message edge W.

KEY RESULT: NYQUIST RATE

$$\begin{aligned} f_s - W &\geq W \\ f_s &\geq 2W \\ f_s^{\min} &= 2W \end{aligned}$$

The least rate $2W$ is the **Nyquist rate**.

The same condition, written for the sampling interval, gives the longest interval allowed. It is the **Nyquist interval**.

NYQUIST INTERVAL

$$\begin{aligned} T_s &= \frac{1}{f_s} \leq \frac{1}{2W} \\ T_s^{\max} &= \frac{1}{2W} \end{aligned}$$

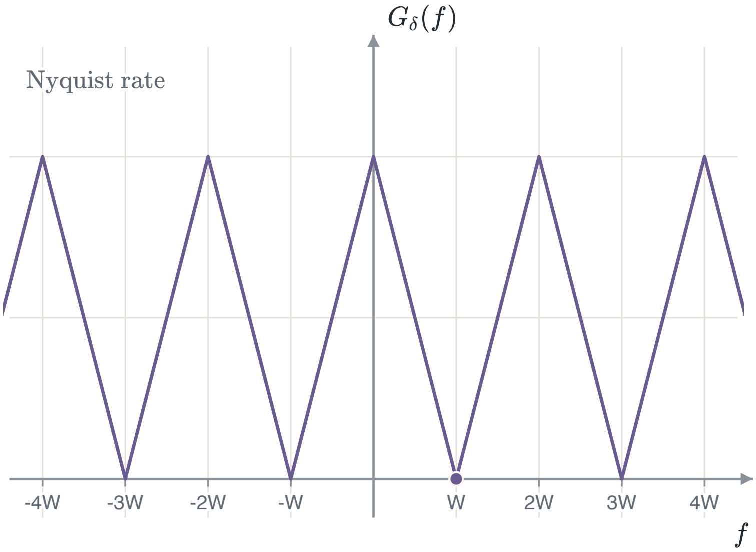


FIGURE 1.6 At $f_s = 2W$ the copies touch without overlap. A lower rate causes overlap. A higher rate leaves a gap.

For example, telephone speech is bandlimited to $W = 3.4$ kHz. The longest sampling interval is $T_s^{\max} = 1/(2W) = 1/(6.8\ \text{kHz}) = 147\ \mu$s.

Aliasing in an image

A camera samples a scene in space with its pixels. A pattern finer than two pixels a cycle folds back into a coarser false pattern, called **moiré**.

Two pixels a cycle is the Nyquist rate in space. An image N pixels across therefore holds a pattern of spatial frequency f only up to a limit.

NO ALIASING

$$f \leq \frac{N}{2} \quad \text{cycles per image width}$$

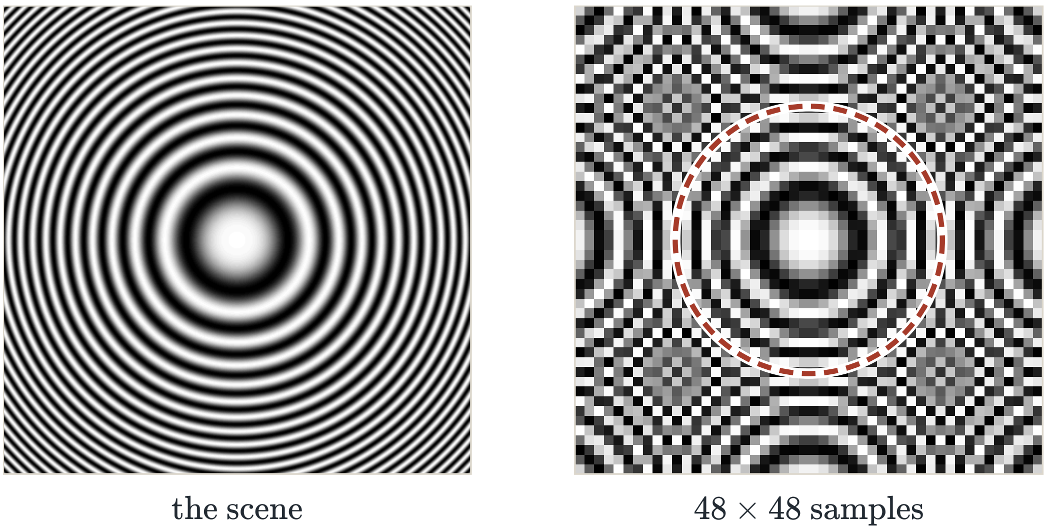


FIGURE 1.7 Rings that get finer toward the edge, and the same rings sampled on a 48×48 grid. Outside the red circle the rings are finer than two pixels a cycle, and false rings appear.

> ⚠ OPTICAL FILTER
>
> Many cameras put a slight blur in front of the sensor. It removes detail finer than the pixels, as an anti-aliasing filter does before a sampler.

For example, an image 1000 pixels wide holds at most $N/2 = 1000/2 = 500$ cycles across its width without aliasing.

Sampling rates around us

> ⓘ RATE FROM THE BAND
>
> Each system samples above twice its highest frequency. A filter before the sampler removes anything above $f_s/2$.

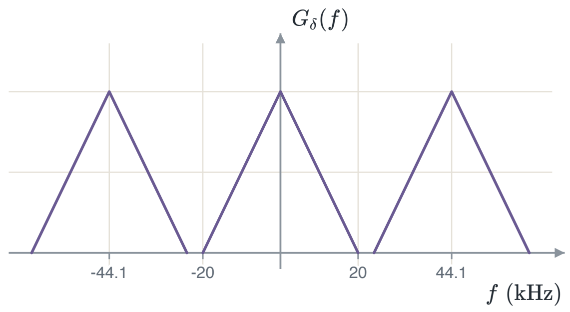


FIGURE 1.8 CD audio is sampled at 44.1 kHz. Hearing ends near 20 kHz, so a 4.1 kHz guard band remains.

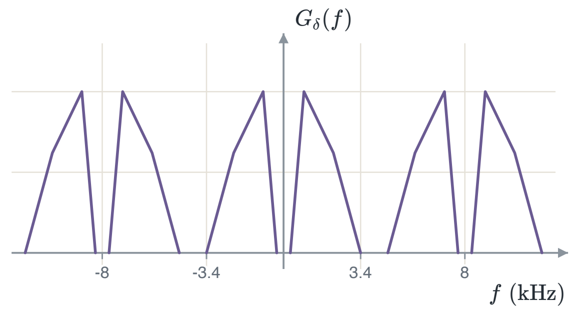


FIGURE 1.9 Telephone speech is filtered to 3.4 kHz and sampled at 8 kHz, which leaves a 1.2 kHz guard band.

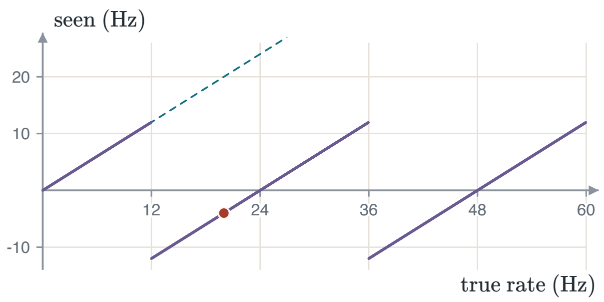


FIGURE 1.10 Film samples at 24 frames a second. A spoke pattern repeating 20 times a second appears to turn backwards at 4 a second.

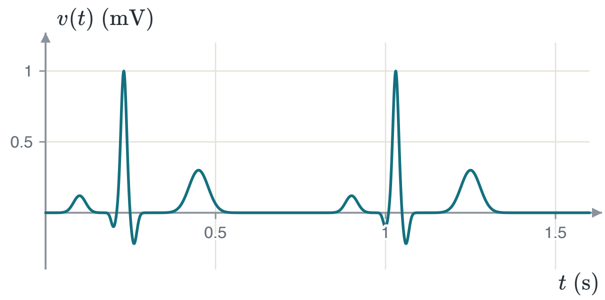


FIGURE 1.11 An electrocardiograph samples at 500 Hz. The diagnostic band ends near 150 Hz, below $f_s/2 = 250$ Hz.

> ⊗ ALIASING YOU CAN SEE
>
> A film camera has no filter before its sampler. A motion faster than half the frame rate folds back, so a wheel can appear to turn backwards.

1.2 Reconstruction

The reconstruction filter

To recover $g(t)$, keep the copy at the origin and reject all the others. An ideal lowpass filter does this.

IDEAL LOWPASS FILTER

$$H(f) = \begin{cases} T_s, & |f| \le W \\ 0, & |f| > f_s - W \end{cases}$$

Between W and $f_s - W$ there is no signal energy, so the filter may do anything there.

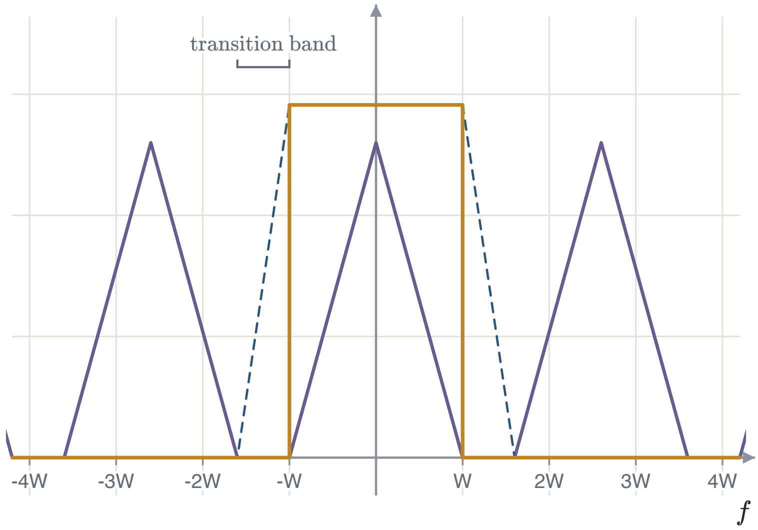


FIGURE 1.12 At $f_s = 2.6W$ the ideal filter (solid, amber) keeps the copy at the origin. A filter that can be built (dashed) falls through the gap before the next copy.

⚠ FILTER GAIN

Sampling scales the spectrum by f_s. A gain of $T_s = 1/f_s$ removes the scale. A unit-gain filter returns f_s times the signal.

ⓘ TRANSITION BAND

Above the Nyquist rate a gap of $f_s - 2W$ opens between W and the next copy. A real filter needs this gap to roll off. The gap is the **transition band**.

For example, let $W = 4$ kHz and $f_s = 10$ kHz. The next copy starts at $f_s - W = 6$ kHz. The transition band is $6 - 4 = 2$ kHz wide.

The reconstruction filter in the time domain

At $f_s = 2W$ the gain is $T_s = 1/(2W)$. Take the inverse transform of the rectangle. Integrate, then use $e^{jx} - e^{-jx} = 2j \sin x$.

IMPULSE RESPONSE AT $f_s = 2W$

$$
\begin{aligned}
h(t) &= \int_{-W}^{W} \frac{1}{2W} e^{j2\pi f t}\, df \\
&= \frac{1}{2W} \cdot \frac{e^{j2\pi W t} - e^{-j2\pi W t}}{j2\pi t} \\
&= \frac{1}{2W} \cdot \frac{2j \sin(2\pi W t)}{j2\pi t} \\
&= \frac{\sin(2\pi W t)}{2\pi W t} \\
&= \operatorname{sinc}(2Wt)
\end{aligned}
$$

The last step uses the sinc convention of the course.

THE SINC CONVENTION

$$
\operatorname{sinc}(x) = \frac{\sin(\pi x)}{\pi x}
$$

The function is one at $x = 0$ and zero at every non-zero integer.

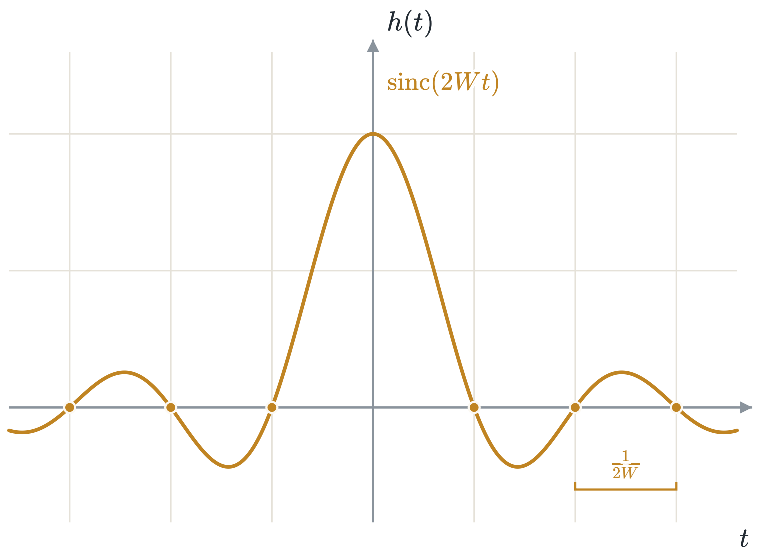


FIGURE 1.13 The pulse $\operatorname{sinc}(2Wt)$ is one at $t = 0$ and zero at every non-zero multiple of $1/(2W)$. These are the sampling instants at the Nyquist rate.

For example, let the message be bandlimited to $W = 5$ kHz. Then $\operatorname{sinc}(2Wt) = 0$ first at $2Wt = 1$. The first zero for $t > 0$ is at $t = 1/(2W) = 100\ \mu$s.

The interpolation formula

The reconstructed signal $g_r(t)$ is the output of the filter when the input is $g_\delta(t)$. Filtering is convolution in time, with τ as the integration variable.

FILTER THE IMPULSE TRAIN

$$
\begin{aligned}
g_r(t) &= \int_{-\infty}^{\infty} \underbrace{\sum_n g(nT_s)\,\delta(\tau - nT_s)}_{g_\delta(\tau)}\, h(t-\tau)\, d\tau \\
&= \sum_n g(nT_s) \int_{-\infty}^{\infty} h(t-\tau)\,\delta(\tau - nT_s)\, d\tau \\
&= \sum_n g(nT_s)\, h(t - nT_s)
\end{aligned}
$$

The weights $g(nT_s)$ do not depend on τ, so the sum and the integral swap. Sifting then sets $\tau = nT_s$ in each term.

Substitute $h(t) = \mathrm{sinc}(2Wt)$, the filter at $f_s = 2W$, with $\mathrm{sinc}(x) = \sin(\pi x)/(\pi x)$.

KEY RESULT: INTERPOLATION FORMULA

$$
g_r(t) = \sum_{n=-\infty}^{\infty} g(nT_s)\, \mathrm{sinc}\big(2W(t - nT_s)\big)
$$

Each sample scales one sinc pulse centred on its own sampling instant.

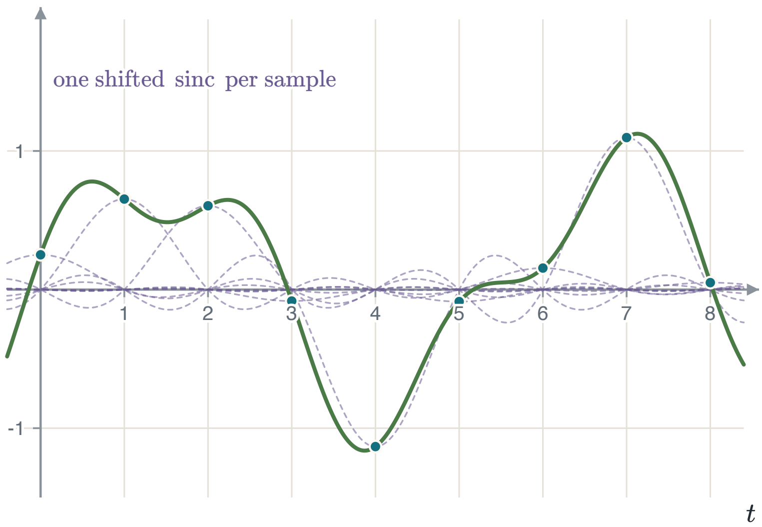


FIGURE 1.14 Each sample (cyan) scales one sinc pulse (dashed). The sum (green) passes through every sample. Here $T_s = 1$ and $2W = 1$.

For example, sample at the Nyquist rate and evaluate $g_r(t)$ at $t = 3T_s$. The term n is $g(nT_s)\,\mathrm{sinc}(3 - n)$. The sinc vanishes at every non-zero integer, so only the term $n = 3$ is non-zero.

Partial sums of the interpolation formula

At the Nyquist rate put $T_s = 1/(2W)$. Then $2WnT_s = n$, and the formula takes a shorter form.

AT THE NYQUIST RATE

$$g(t) = \sum_n g(nT_s)\,\mathrm{sinc}\big(2W(t - nT_s)\big)$$
$$= \sum_{n=-\infty}^{\infty} g\left(\frac{n}{2W}\right)\mathrm{sinc}(2Wt - n)$$

VALUES AT THE SAMPLE TIMES

$h(nT_s) = \mathrm{sinc}(n) = 0$ for $n \neq 0$. So at $t = kT_s$ only the term $n = k$ survives, and $g_r(kT_s) = g(kT_s)$.

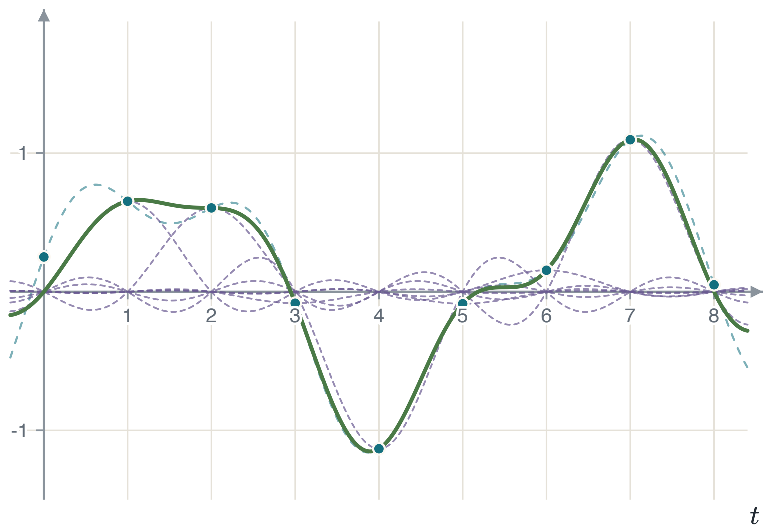


FIGURE 1.15 The partial sum of the terms $n = 1, \ldots, 7$ (green) against $g(t)$ (dashed). Each new sinc pulse corrects the sum between the samples without moving it at the samples already fitted.

A FINITE SUM

A sum over a few samples is exact only at those samples. The error between them shrinks as more terms enter.

For example, let only one sample be non-zero: $g(0) = 1$. The sum has one term, so the formula reconstructs $1 \cdot \mathrm{sinc}(2Wt)$.

EXAMPLE 1.1 — THE NYQUIST RATE, A GUARD BAND AND A MODULATED SIGNAL

GIVEN $x(t)$ is bandlimited to $W = 40$ kHz, and $y(t) = x(t)\cos(80000\pi t)$.

FIND (a) The Nyquist rate of $x(t)$. (b) The rate of $x(t)$ with a 10 kHz guard band. (c) The Nyquist rate of $y(t)$.

METHOD The Nyquist rate is $2W$, and a guard band f_g adds to it: $f_s = 2W + f_g$. For (c), find the highest frequency of y first. Modulation shifts the spectrum, so the rate follows the new band edge.

SOLUTION (a) and (b) follow from the two rules.

$$
\text{(a)} \quad \begin{aligned} f_s &= 2W \\ &= 2(40) \\ &= 80 \text{ kHz} \end{aligned} \qquad \text{(b)} \quad \begin{aligned} f_s &= 2W + f_g \\ &= 80 + 10 \\ &= 90 \text{ kHz} \end{aligned}
$$

(c) The carrier frequency comes from $2\pi f_c t = 80000\pi t$, so $f_c = 40$ kHz. Write the cosine as two exponentials. Each exponential $e^{\pm j2\pi f_c t}$ shifts the spectrum by $\pm f_c$.

$$
\begin{aligned} y(t) &= x(t)\cos(2\pi f_c t) \\ &= \tfrac{1}{2}x(t)\, e^{j2\pi f_c t} + \tfrac{1}{2}x(t)\, e^{-j2\pi f_c t} \\ Y(f) &= \tfrac{1}{2}X(f - f_c) + \tfrac{1}{2}X(f + f_c) \end{aligned}
$$

The highest frequency of y and its Nyquist rate follow.

$$
\begin{aligned} f_{\max} &= f_c + W \\ &= 40 + 40 \\ &= 80 \text{ kHz} \end{aligned} \qquad \begin{aligned} f_s &= 2f_{\max} \\ &= 2(80) \\ &= 160 \text{ kHz} \end{aligned}
$$

CHECK

At 90 kHz the first copy spans $f_s - W = 50$ to $f_s + W = 130$ kHz. The gap from $W = 40$ to 50 kHz is the 10 kHz guard band. At 80 kHz the first copy starts exactly at the message edge $W = 40$ kHz.

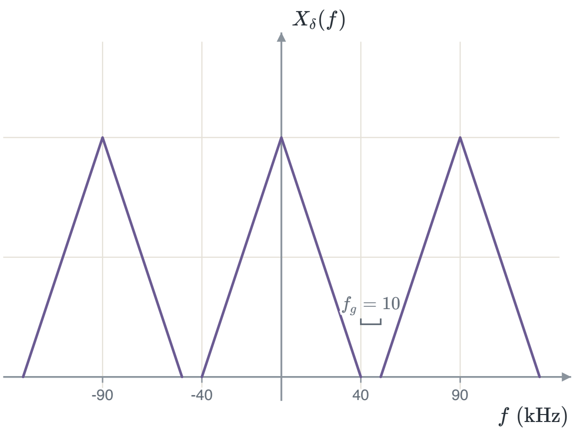


FIGURE 1.16 Parts (a) and (b): the sampled spectrum at $f_s = 90$ kHz. The first copy starts at 50 kHz and leaves a 10 kHz guard band. At $f_s = 80$ kHz it would start at 40 kHz.

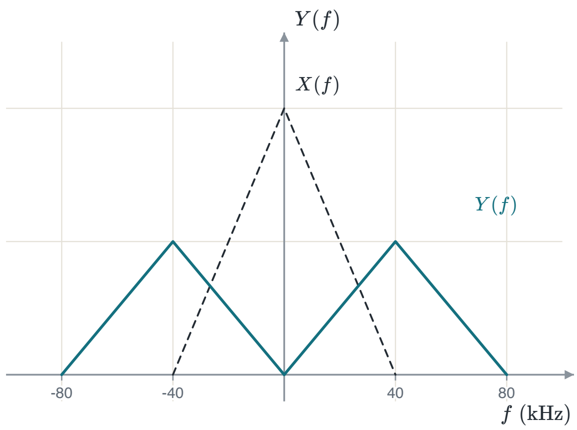


FIGURE 1.17 Part (c): $X(f)$ (dashed) and the two half-height copies that make $Y(f)$. The band edge of $Y(f)$ is 80 kHz.

COMMON ERROR

The rate 80 kHz belongs to x. Modulation moves the highest frequency to 80 kHz, so y needs 160 kHz.

Reconstruction around us

> ⓘ A HOLD, NOT A SINC
>
> A converter cannot build the ideal filter. It holds each sample for T_s, then smooths the staircase with an analog filter.

A hold of length T_s has a rectangular impulse response, and a rectangle transforms to a sinc. Here $\operatorname{sinc}(x) = \sin(\pi x)/(\pi x)$.

> TRANSFORM PAIR
>
> $$h_0(t) = 1,\ 0 \le t < T_s \ \leftrightarrow\ T_s \operatorname{sinc}(fT_s)\, e^{-j\pi f T_s}$$

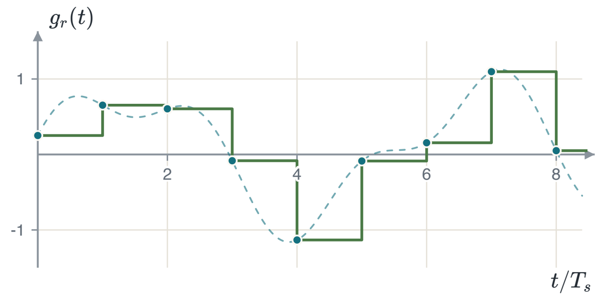


FIGURE 1.18 A zero-order hold keeps each sample until the next one. The output is a staircase.

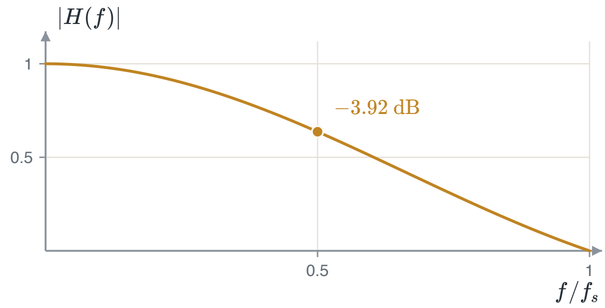


FIGURE 1.19 The hold is a filter with gain $|\operatorname{sinc}(f/f_s)|$. It falls to $2/\pi$ at $f_s/2$.

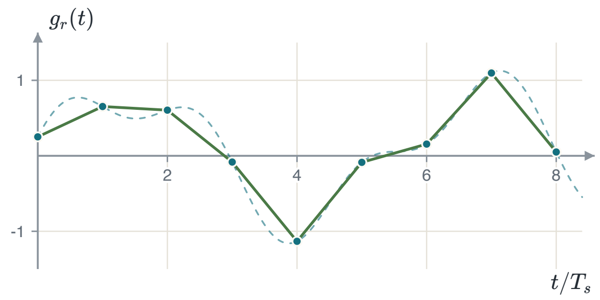


FIGURE 1.20 Joining the samples by straight lines is a filter with a triangular impulse response. It is smoother than the hold, but still not a sinc.

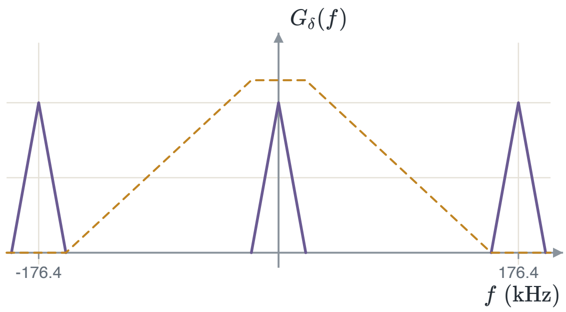


FIGURE 1.21 Oversampling CD audio by four moves the first copy to 156.4 kHz. A gentle filter (dashed) then removes it.

> ⚠ HOLD DROOP
>
> The hold weights the spectrum by $\operatorname{sinc}(f/f_s)$. At $f_s/2$ that is $2/\pi$, a loss of 3.92 dB.

1.3 Quantization

Uniform quantization

> ⓘ QUANTIZATION
>
> Quantization replaces each sample by the nearest of L **representation levels**. In a **uniform** quantizer the levels are equally spaced.

The L levels share the full range $[-m_{\max}, m_{\max}]$ equally. The spacing is the **step size** Δ.

> STEP SIZE
>
> $$\Delta = \frac{2m_{\max}}{L}$$
>
> For a range $[m_{\min}, m_{\max}]$ the step is $\Delta = (m_{\max} - m_{\min})/L$.

> ⓘ MID-RISE
>
> A decision boundary sits at zero. With L even, the levels are $\pm\Delta/2, \pm 3\Delta/2, \ldots$ and none of them is zero.

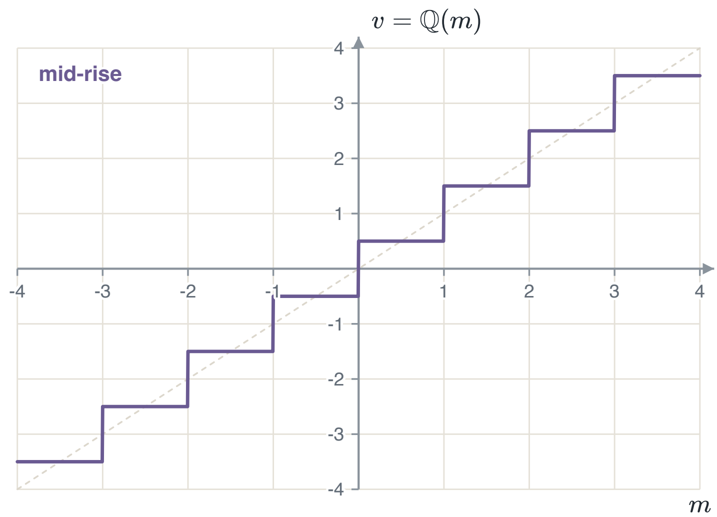


FIGURE 1.22 A mid-rise quantizer with $L = 8$ and $\Delta = 1$. A boundary sits at the origin, so the output steps through $\pm\Delta/2$.

For example, a mid-rise quantizer with $\Delta = 1$ maps $m = 0.2$ to 0.5. The input lies in the region $(0, 1]$, whose level is its midpoint 0.5.

Mid-rise and mid-tread

Uniform quantizers come in two families. They differ in what they do at zero.

TABLE 1.1 The two families of uniform quantizer.

FAMILY	AT ZERO	A SMALL INPUT
Mid-rise	a boundary	goes to $\pm\Delta/2$
Mid-tread	a level	goes to exactly 0

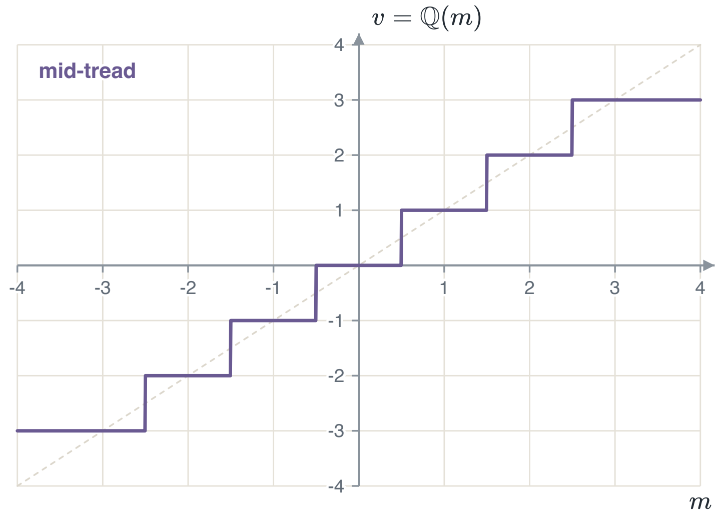


FIGURE 1.23 A mid-tread quantizer with $\Delta = 1$. A level sits at the origin, and the tread around zero is one step wide.

> ⚠ SILENCE
>
> A mid-rise quantizer turns a silent input with a little noise into a signal of $\pm\Delta/2$. A mid-tread quantizer keeps it silent.

For example, a mid-tread quantizer with $\Delta = 1$ maps $m = 0.2$ to 0. The tread around zero covers $|m| < 0.5$, and its level is 0.

A sampled signal through the quantizer

A converter samples first and quantizes second. Each sample passes through three steps.

1. Sample: read $m(nT_s)$ off the signal.
2. Find the region: the sample lies between two boundaries of the staircase.
3. Output: the level of that region, $v[n] = \mathbb{Q}\big(m(nT_s)\big)$.

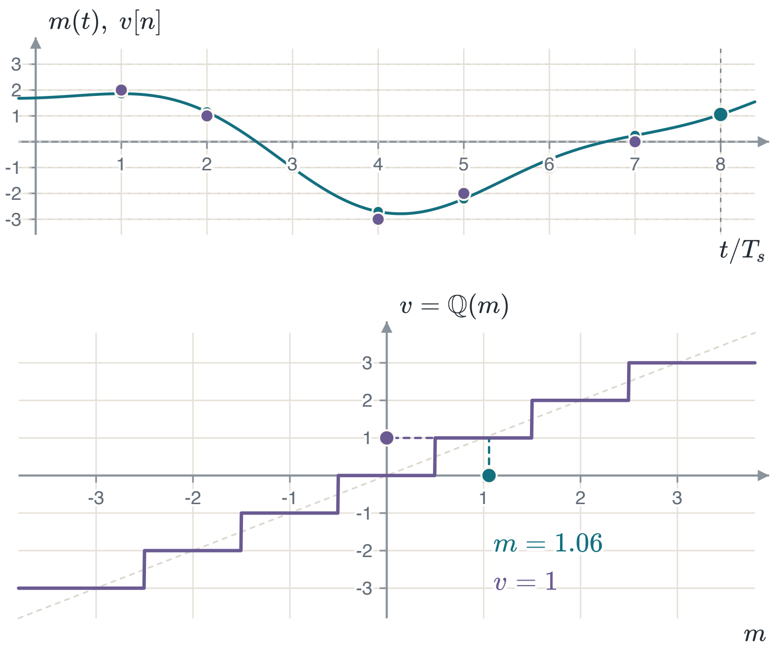


FIGURE 1.24 Eight samples $m(nT_s)$ (cyan) enter the mid-tread quantizer with $\Delta = 1$ and leave as the levels $v[n]$ (violet). The lower panel follows the eighth sample across the staircase.

> ⚠ MANY INPUTS, ONE LEVEL
>
> Every input between 1.5 and 2.5 leaves as 2. The output keeps the region and loses the exact value.

For example, the same quantizer maps the sample $m(nT_s) = 1.62$ to $v[n] = 2$. The sample lies between the boundaries 1.5 and 2.5.

The quantizer as a function

A quantizer is a function of its input. Its **boundaries** m_k cut the range into L regions. Region k is the interval $m_{k-1} < m \le m_k$, and its level is v_k.

QUANTIZER FUNCTION

$$\mathbb{Q}(m) = v_k \quad \text{for} \quad m_{k-1} < m \le m_k$$

An optimal quantizer meets two conditions.

1. Each boundary is the midpoint of its two levels: $m_k = \frac{1}{2}(v_k + v_{k+1})$. Every input then goes to the nearer level.
2. Each level is the centroid of its region: $v_k = E[M \mid M \in \text{region } k]$.

A quantizer that meets both conditions is the **Lloyd–Max quantizer**.

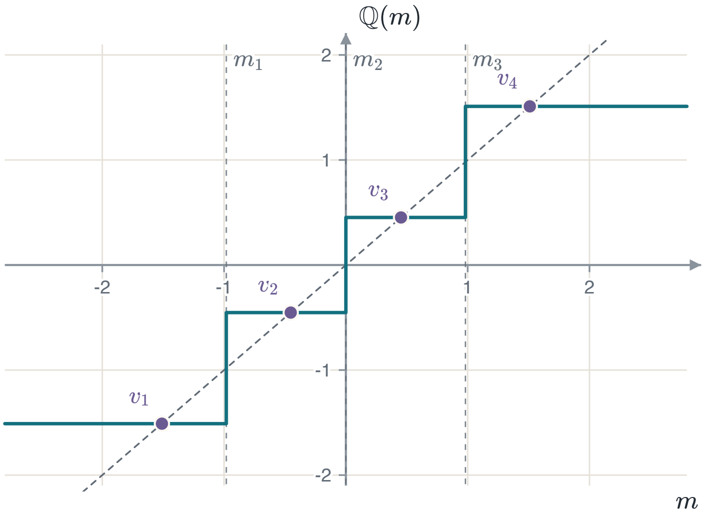


FIGURE 1.25 The best four-level quantizer for a Gaussian input. The steps widen in the tails, where the input is rare. Each boundary m_k lies midway between two levels.

> UNIFORM INPUT
>
> A uniform quantizer always meets the midpoint condition. For a uniform input it meets the centroid condition too, so it is optimal.

For example, let two neighbouring levels be $v_1 = 1$ and $v_2 = 3$. The midpoint condition puts the boundary between them at $\frac{1}{2}(1+3) = 2$.

Overload and granular noise

A quantizer spanning $[-m_{\max}, m_{\max}]$ makes two kinds of error. The choice of $m_{\max}$ trades one against the other.

TABLE 1.2 The two errors of a quantizer.

ERROR	WHERE	SIZE
Overload	an input beyond $\pm m_{\max}$, clipped to the outer level	no bound
Granular noise	an input inside the range	$\lvert q\rvert \le \Delta/2$, larger for a wider range

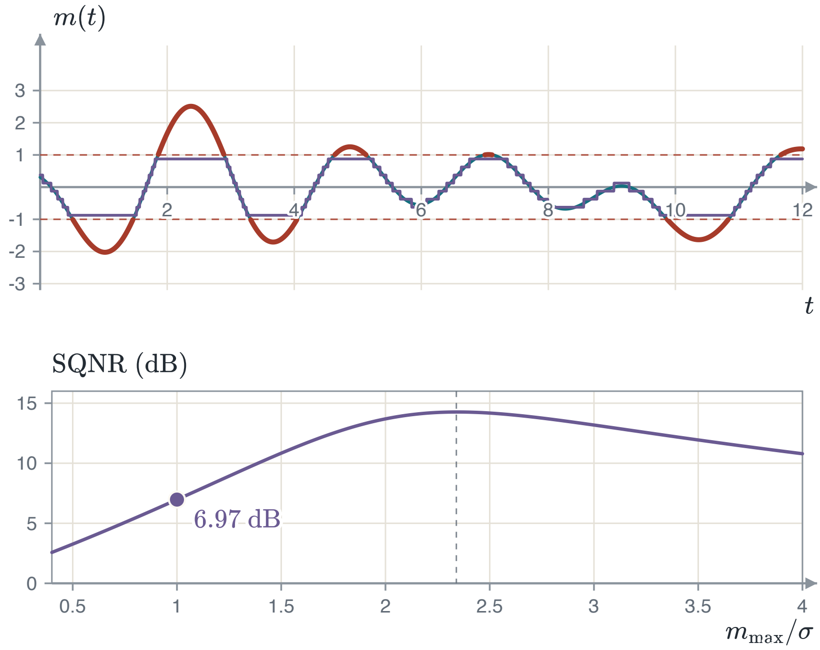


FIGURE 1.26 A signal with rare large peaks through a 3-bit quantizer spanning $[-m_{\max}, m_{\max}]$ with $m_{\max} = \sigma$. Red marks the clipped peaks. The lower curve is the SQNR of a Gaussian input against the range, and the dot sits at $m_{\max} = \sigma$.

THE BEST RANGE

For a Gaussian input and 3 bits the SQNR peaks at $m_{\max} = 2.34\sigma$, with 14.27 dB. A few clipped peaks cost less than a coarser step.

COMMON ERROR

Set $m_{\max}$ from the rare peaks of the signal, not from its typical size. At $m_{\max} = \sigma$ the SQNR falls to 6.97 dB.

For example, set a 3-bit quantizer to $m_{\max} = \sigma$ for a Gaussian input. About 32% of the samples lie outside $\pm\sigma$, and each of them is clipped. Overload then dominates.

Quantizers around us

EVERY READING IS A LEVEL

A digital display shows one of a finite set of values. A change smaller than one step leaves the reading unchanged.

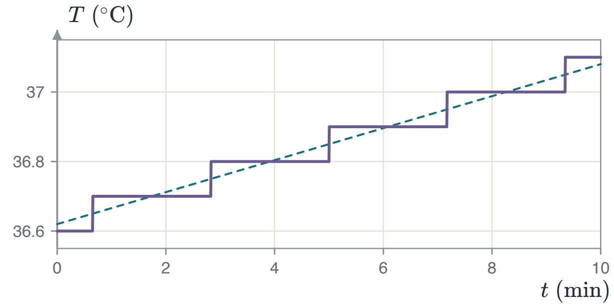


FIGURE 1.27 A clinical thermometer shows 0.1 °C steps. The reading jumps only when the temperature crosses a boundary.

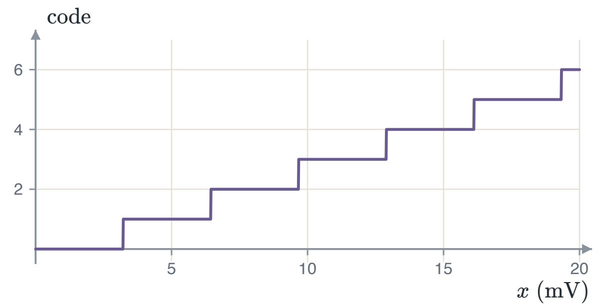


FIGURE 1.28 A 10-bit converter over 3.3 V has $\Delta = 3.3/1024 = 3.22$ mV. Inputs inside one step share one code.

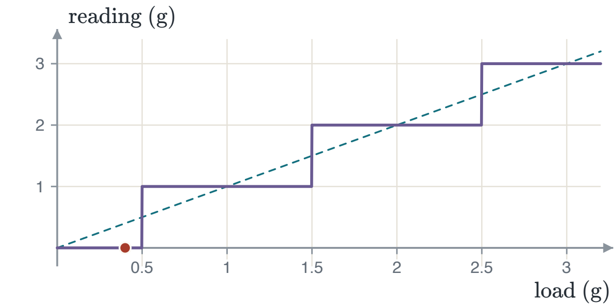


FIGURE 1.29 A kitchen scale reads in 1 g steps and rounds. A 0.4 g feather reads 0: the scale is a mid-tread quantizer.

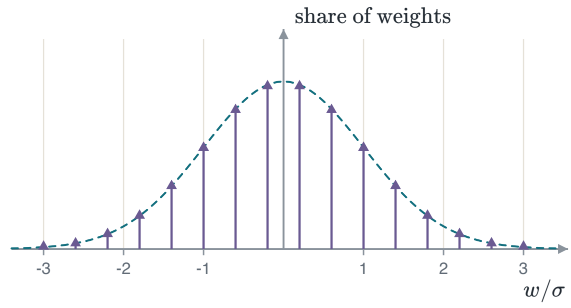


FIGURE 1.30 A language model with 70×10^9 weights needs 140 GB at 16 bits. At 4 bits each weight is rounded to one of $2^4 = 16$ levels, and the model needs about 35 GB.

> A LEVEL AT ZERO
>
> A display that rounds is mid-tread, so an empty scale reads exactly zero. A converter that truncates puts a boundary at each code edge.

1.4 Quantization noise and SQNR

Quantization noise

The **quantization error** is the difference between the input and its level.

QUANTIZATION ERROR

$$q = m - \mathbb{Q}(m)$$

Inside the range, v_k is the midpoint of a region Δ wide. The input cannot lie farther from the midpoint than half the region.

BOUND

$$\begin{aligned}|q| &= |m - v_k| \\ &\le \tfrac{1}{2}(m_k - m_{k-1}) \\ &= \frac{\Delta}{2}\end{aligned}$$

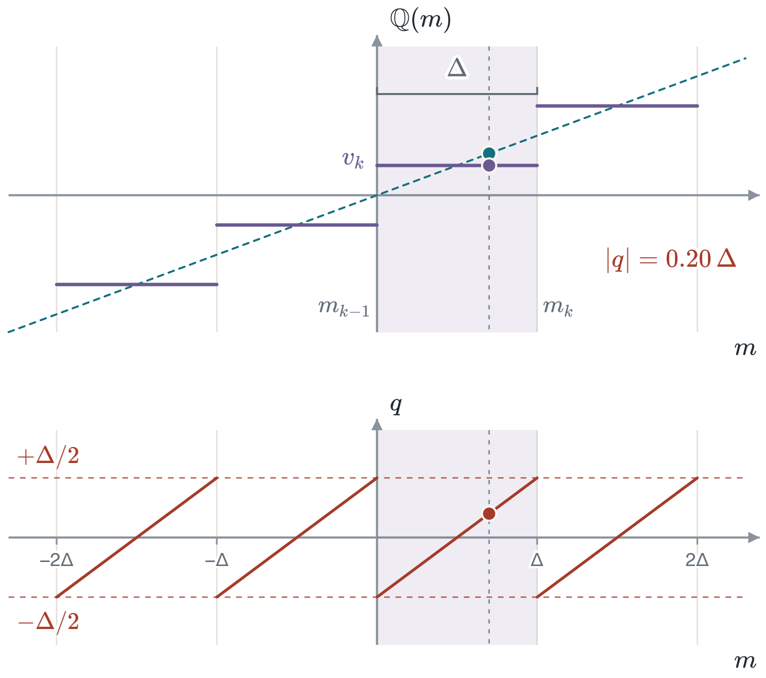


FIGURE 1.31 The error is the gap between the line $v = m$ and the staircase, shown at $m = 0.7\Delta$. It is largest at a boundary, where it equals $\Delta/2$.

ⓘ UNIFORM MODEL

For a small Δ the input density is nearly flat across one region. The error is then modelled as $Q \sim U(-\Delta/2, \Delta/2)$.

For example, a quantizer with $\Delta = 1.25$ whose input stays inside its range has $|q| \le \Delta/2 = 0.625$.

The power of quantization noise

The power of the error is its mean-square value $E[Q^2]$. Under the uniform model the error density is $f_Q(q) = 1/\Delta$ on $[-\Delta/2, \Delta/2]$. Write the expectation as an integral and evaluate the antiderivative at its limits.

MEAN-SQUARE ERROR

$$\begin{aligned} E[Q^2] &= \int_{-\Delta/2}^{\Delta/2} q^2 \, \frac{1}{\Delta} \, dq \\ &= \frac{1}{\Delta} \left[\frac{q^3}{3} \right]_{-\Delta/2}^{\Delta/2} \\ &= \frac{1}{3\Delta} \left(\frac{\Delta^3}{8} + \frac{\Delta^3}{8} \right) \\ &= \frac{\Delta^2}{12} \end{aligned}$$

The lower limit gives $(-\Delta/2)^3 = -\Delta^3/8$, which is subtracted.

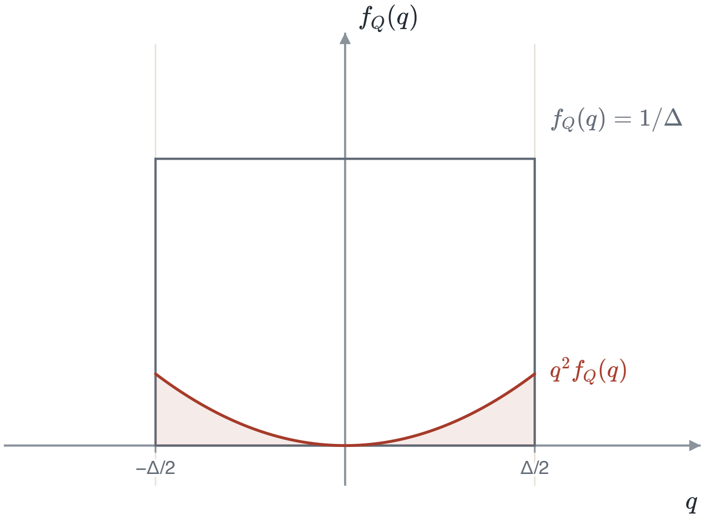


FIGURE 1.32 The uniform error density, and $q^2 f_Q(q)$ shaded. The shaded area is the noise power $E[Q^2]$.

For example, a fine uniform quantizer with $\Delta = 0.5$ has $E[Q^2] = \Delta^2/12 = 0.25/12 = 0.0208$.

Quantization noise in bits

An R-bit quantizer has $L = 2^R$ levels. Substitute $\Delta = 2m_{\max}/L$ and $L = 2^R$ into $E[Q^2] = \Delta^2/12$.

IN BITS

$$\begin{aligned} E[Q^2] &= \frac{\Delta^2}{12} \\ &= \frac{1}{12} \left(\frac{2m_{\max}}{2^R} \right)^2 \\ &= \frac{1}{12} \cdot \frac{4m_{\max}^2}{2^{2R}} \\ &= \frac{m_{\max}^2}{3 \cdot 2^{2R}} \end{aligned}$$

ONE BIT, A QUARTER

Each extra bit halves Δ and divides the noise power by four.

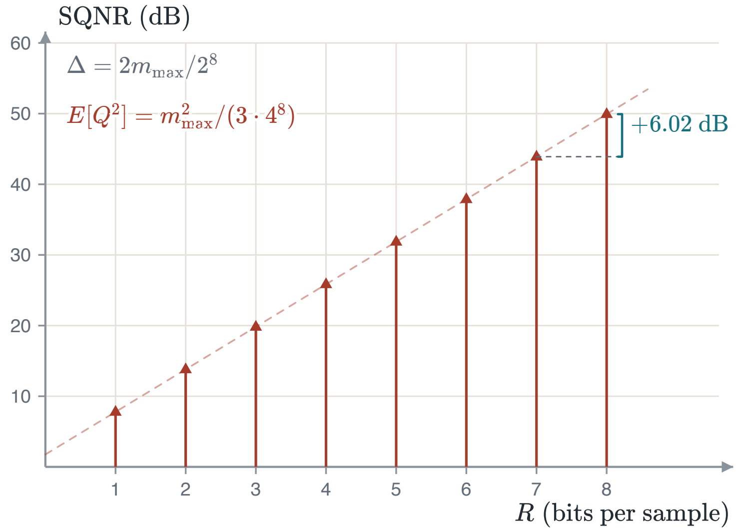


FIGURE 1.33 The SQNR of a full-scale sinusoid for $R = 1, \ldots, 8$. Each added bit halves Δ, divides $E[Q^2]$ by four, and lifts the SQNR by $10\log_{10} 4 = 6.02$ dB.

For example, a fine quantizer that goes from $R = 6$ to $R = 7$ bits divides $E[Q^2]$ by 4. The noise power is proportional to 2^{-2R}, and $2^2 = 4$.

Signal-to-quantization-noise ratio

The **signal-to-quantization-noise ratio** (SQNR) compares the power P_M of the message with the power of the error. Substitute $E[Q^2] = m^2_{\max}/(3 \cdot 2^{2R})$.

UNIFORM QUANTIZER

$$\begin{aligned} \mathrm{SQNR} &= \frac{P_M}{E[Q^2]} \\ &= \frac{3P_M}{m^2_{\max}}\, 2^{2R} \end{aligned}$$

In decibels the product becomes a sum, because the log of a product is a sum of logs. The factor 2^{2R} gives $20R\log_{10} 2$, and $20\log_{10} 2 = 6.02$.

KEY RESULT: SIX DECIBELS A BIT

$$\begin{aligned} \mathrm{SQNR}\ [\mathrm{dB}] &= \underbrace{10\log_{10}\frac{3P_M}{m^2_{\max}}}_{\alpha} + 10\log_{10} 2^{2R} \\ &= \alpha + 20R\log_{10} 2 \\ &= \alpha + 6.02R \end{aligned}$$

Every extra bit adds 6.02 dB.

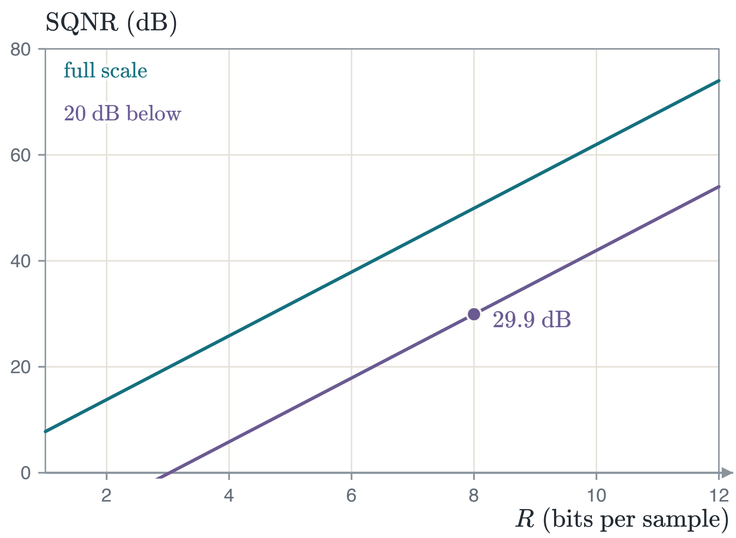


FIGURE 1.34 The SQNR of a full-scale sinusoid (cyan) and of the same sinusoid 20 dB below full scale (violet). The line keeps its slope of 6.02 dB a bit and drops by the level.

For example, a quantizer that gives 40 dB at $R = 6$ gives 52.04 dB at $R = 8$. Two more bits add $2(6.02) = 12.04$ dB.

SQNR of a random variable and of a signal

The SQNR is a ratio of two powers. For a random variable M quantized to $\mathbb{Q}(M)$, both powers are expected values.

RANDOM VARIABLE

$$\text{SQNR} = \frac{E[M^2]}{E\big[(M - \mathbb{Q}(M))^2\big]}$$

For a signal $m(t)$ quantized to $\mathbb{Q}(m(t))$, both powers are time averages over the whole time axis.

SIGNAL

$$\text{SQNR} = \frac{\lim\limits_{T\to\infty} \dfrac{1}{T} \displaystyle\int_{-T/2}^{T/2} m^2(t)\, dt}{\lim\limits_{T\to\infty} \dfrac{1}{T} \displaystyle\int_{-T/2}^{T/2} \big(m(t) - \mathbb{Q}(m(t))\big)^2\, dt}$$

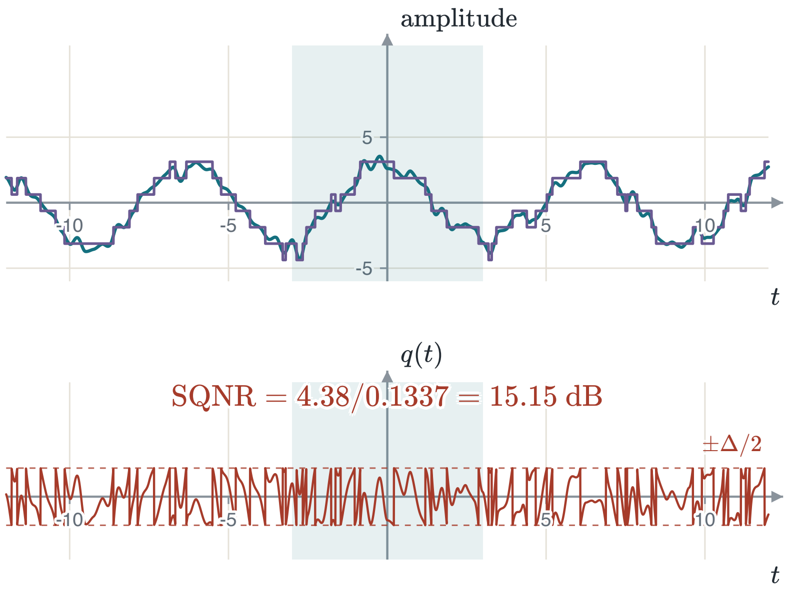


FIGURE 1.35 A sinusoid with Gaussian noise on it, through a 3-bit quantizer spanning $[-5, 5]$, averaged over the window $T = 6$ s (shaded). The ratio of the two time averages settles as T grows.

> THE NOISY SINUSOID
>
> Here $m(t) = 3\cos t + n(t)$, with $n(t)$ Gaussian of variance 0.36. As $T \to \infty$ the numerator tends to $P_M = 4.5 + 0.36 = 4.86$. The uniform error model puts the denominator at $\Delta^2/12$.

EXAMPLE 1.2 — SQNR OF A SINUSOID

GIVEN $m(t) = 5\cos t$, and a uniform quantizer over its full range with $R = 3$.

FIND The step size and the SQNR.

METHOD Find the signal power and the step size. The noise power is $\Delta^2/12$, and the SQNR is the ratio of the two powers. A sinusoid of peak A has power $A^2/2$.

SOLUTION The range $[-5, 5]$ is 10 wide, and $R = 3$ gives $L = 2^3 = 8$ levels.

$$P_M = \tfrac{1}{2}(5)^2 = 12.5$$
$$\Delta = \frac{2(5)}{2^3} = 1.25$$

Divide the signal power by the noise power.

$$E[Q^2] = \frac{1.25^2}{12} = 0.1302$$
$$\mathrm{SQNR} = 10\log_{10}\frac{12.5}{0.1302} = 19.82\ \mathrm{dB}$$

CHECK The rule gives the same value: $\alpha = 10\log_{10} 1.5 = 1.76$ dB and $1.76 + 6.02(3) = 19.82$ dB. Measured on the waveform the SQNR is 19.09 dB, since three bits is a coarse quantizer.

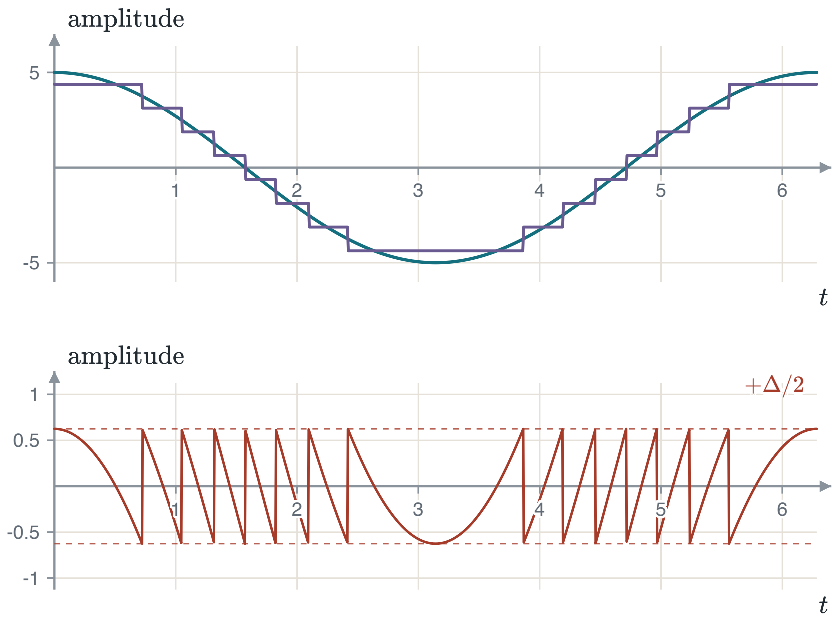


FIGURE 1.36 $m(t) = 5\cos t$ through a 3-bit quantizer spanning $[-5, 5]$, and the error below it. The error stays between $-\Delta/2$ and $\Delta/2$.

> ⊗ COMMON ERROR
>
> Use $\Delta = 2m_{\max}/L$, not $m_{\max}/L$. The full range is $2m_{\max} = 10$ V.

EXAMPLE 1.3 — SQNR OF A UNIFORM SOURCE

GIVEN $M \sim U(-1, 1)$, and a uniform quantizer with $L = 256$.

FIND The SQNR.

METHOD Compute the two powers. The signal power comes from the density $f_M(m) = \frac{1}{2}$ on $[-1, 1]$. The noise power is $\Delta^2/12$ with $\Delta = 2/256 = 1/128$.

SOLUTION Integrate $m^2 f_M(m)$ and evaluate the antiderivative at its limits.

$$P_M = \int_{-1}^{1} \tfrac{1}{2} m^2 \, dm = \tfrac{1}{2}\left[\frac{m^3}{3}\right]_{-1}^{1} = \tfrac{1}{2} \cdot \tfrac{2}{3} = \tfrac{1}{3}$$

Then the noise power and the ratio follow.

$$E[Q^2] = \frac{(1/128)^2}{12} = \frac{1}{196\,608}$$

$$\begin{aligned} \mathrm{SQNR} &= \tfrac{1}{3}(196\,608) \\ &= 65\,536 = 2^{16} \\ &= 48.16 \text{ dB} \end{aligned}$$

CHECK Each region holds a flat piece of the input density, so the error is exactly $U(-\Delta/2, \Delta/2)$. The result is $6.02(8)$ dB, so $\alpha = 0$ for this source.

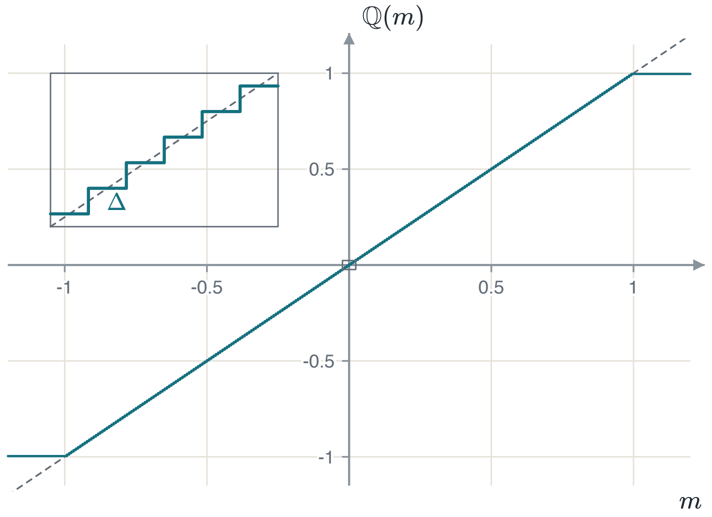


FIGURE 1.37 The 256-level quantizer on $[-1, 1]$. At full scale the steps are too fine to see. The inset enlarges the six around the origin.

> ⊗ COMMON ERROR
>
> Use $\Delta = 2/256$, not $1/256$. The range $[-1, 1]$ is 2 wide.

EXAMPLE 1.4 — SQNR OF A GAUSSIAN SOURCE

GIVEN A zero-mean stationary Gaussian source has $S_X(f) = 2$ for $|f| < 100$ Hz. Its samples enter a quantizer with five levels $0, \pm 10, \pm 30$ and boundaries $\pm 20, \pm 40$.

FIND The SQNR.

METHOD The signal power is the area under the power spectral density. The quantizer is coarse and its outer regions are unbounded, so the noise power is integrated region by region.

SOLUTION The signal power is the area under S_X, not its height.

$$P_X = \int_{-100}^{100} 2\,df = 2(200) = 400$$

The mean is zero, so 400 is also the variance, and f_X is Gaussian with $\sigma = 20$. The density f_X is even, so each side region has a twin. The integrals are evaluated numerically.

$$\begin{aligned} P_Q &= \int_{-20}^{20} x^2 f_X\,dx + 2\int_{20}^{40} (x-10)^2 f_X\,dx + 2\int_{40}^{\infty} (x-30)^2 f_X\,dx \\ &= 79.50 + 2(46.36) + 2(7.98) = 188.17 \\ \mathrm{SQNR} &= 10\log_{10}\frac{400}{188.17} = 3.28\text{ dB} \end{aligned}$$

CHECK The five areas of the error integrand add to $79.50 + 92.72 + 15.96 = 188.18$, which is 188.17 before rounding. The ratio $400/188.17 = 2.126$, and $10\log_{10} 2.126 = 3.28$ dB.

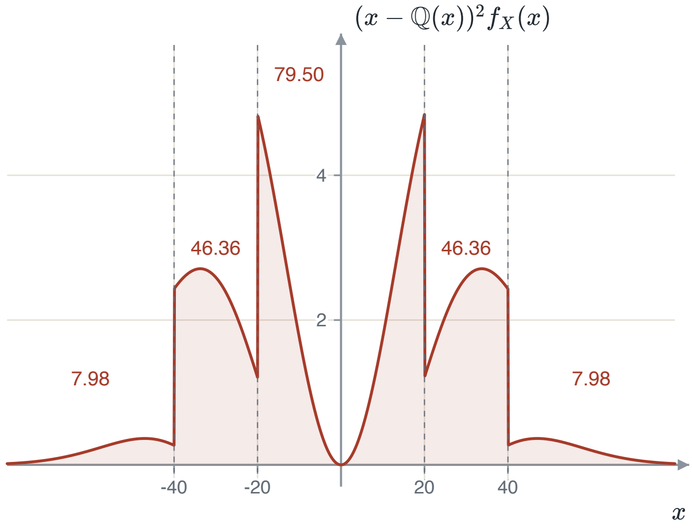


FIGURE 1.38 The error integrand $(x-\mathbb{Q}(x))^2 f_X(x)$, region by region. The five areas add to the noise power $P_Q = 188.17$.

> ⊗ COMMON ERROR
>
> Using $\Delta^2/12 = 33.3$ gives 10.8 dB, 7.5 dB too high. That model holds only for a small Δ and an input inside the range.

One rule, three sources

Every source gains 6.02 dB a bit. The source sets only the intercept, $\alpha = 10\log_{10}(3P_M/m_{\max}^2)$. Three sources at full range give three intercepts.

> THREE INTERCEPTS
>
> $$\begin{aligned} \text{sinusoid:}\quad & 10\log_{10}\frac{3(A^2/2)}{A^2} = 1.76\text{ dB} \\ \text{uniform:}\quad & 10\log_{10}\frac{3(1/3)}{1} = 0\text{ dB} \\ \text{Gaussian, } m_{\max} = 4\sigma\text{:}\quad & 10\log_{10}\frac{3\sigma^2}{16\sigma^2} = -7.27\text{ dB} \end{aligned}$$

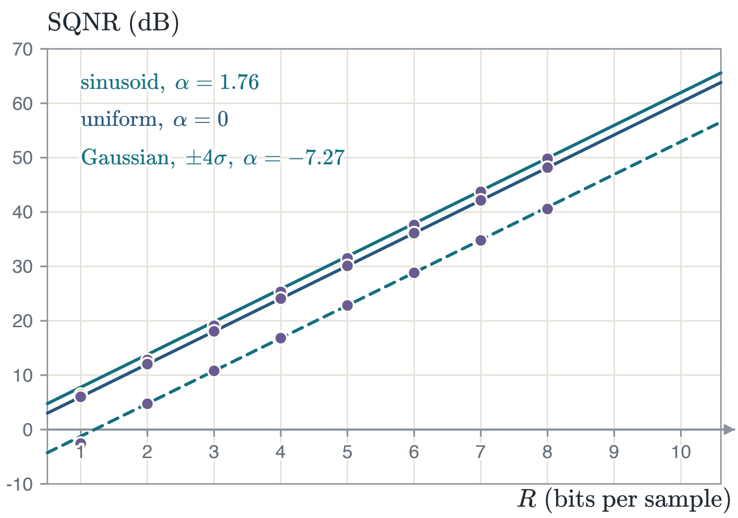


FIGURE 1.39 The lines are $\alpha + 6.02R$ for the three sources. The dots are measured on each quantized source, up to $R = 8$. The Gaussian dots leave their line at many bits, where the clipped peaks set the error.

> ⚠ PEAKS COST RANGE
>
> A Gaussian source has rare large peaks. A range wide enough for them leaves most levels unused on typical samples.

For example, at $R = 8$ the rule puts the Gaussian source at $\pm 4\sigma$ below the sinusoid by $1.76 - (-7.27) = 9.03$ dB. The slopes are equal, so the gap is the gap in α.

Hearing quantization noise

Speech quantized with 8 bits has an error that sounds like a faint hiss. At 2 bits the error follows the words.

> ⚠ NOISE THAT FOLLOWS THE SIGNAL
>
> At a few bits the error moves with the speech. It sounds like distortion, not like hiss, and the uniform error model no longer holds.

For example, speech that sounds clean at 8 bits loses $4(6.02) = 24.08$ dB of SQNR at 4 bits, by the rule of six decibels a bit.

Dither

> ⓘ DITHER
>
> **Dither** is a small random signal, about one step wide, added before the quantizer. The error then no longer follows the signal.

> ✓ THE AVERAGE FOLLOWS
>
> Each output still sits on a level. Averaged over time, or by the eye over neighbouring pixels, it follows the input between the levels.

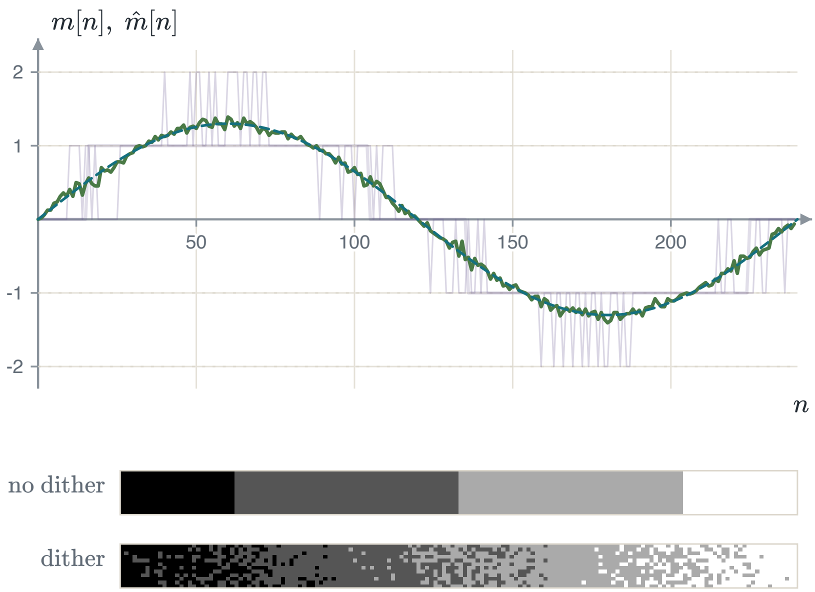


FIGURE 1.40 A slow sinusoid 1.3 steps high through a quantizer with a unit step. The faint traces are the staircase without dither and one output with dither. The green trace averages 64 dithered outputs. The strips are a grey ramp at four levels, without and with dither.

> ⚠ THE COST
>
> Dither adds noise power. It trades a pattern that the ear or the eye notices for a hiss or a grain that it ignores.

For example, let an input sit 0.3Δ above a level, with dither uniform on $[-\Delta/2, \Delta/2]$. The boundary is 0.5Δ up, so the dither must exceed 0.2Δ. That happens with probability 0.3, so the next level up is chosen 30% of the time. The average output is then 0.3Δ above the level.

SQNR around us

> ⓘ HEADROOM COSTS SQNR
>
> The formula assumes a full-scale input. Each decibel below full scale takes one decibel from the SQNR.

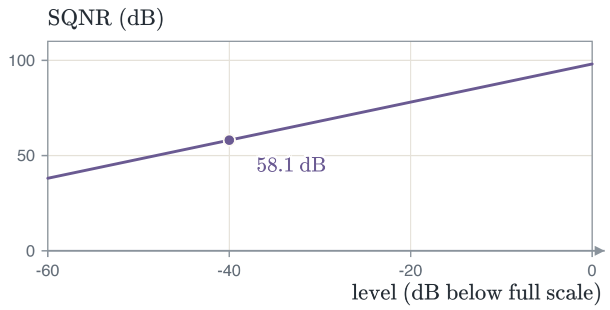


FIGURE 1.41 CD audio uses 16 bits: 98.1 dB at full scale, and 58.1 dB for a passage 40 dB quieter.

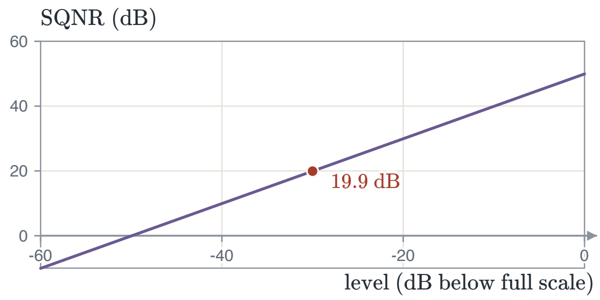


FIGURE 1.42 By $\alpha + 6.02R$, uniform 8-bit PCM gives 49.9 dB at full scale but only 19.9 dB for a signal 30 dB below it.

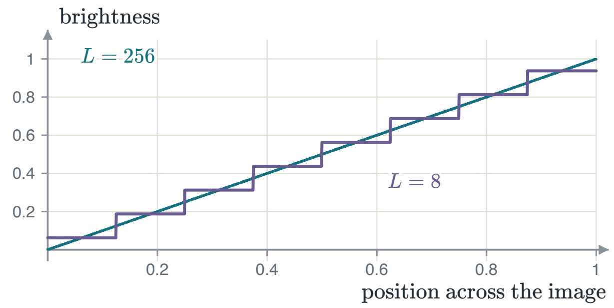


FIGURE 1.43 A row of an image at 8 levels shows bands where the 256-level row is smooth.

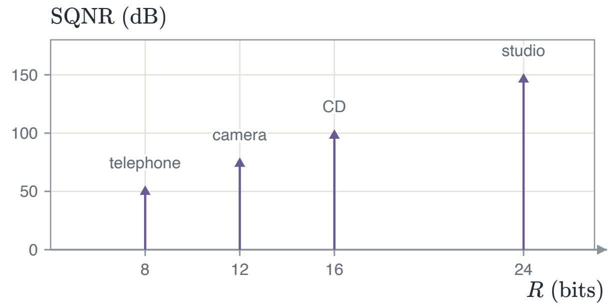


FIGURE 1.44 A full-scale sinusoid at common word lengths: 49.9, 74.0, 98.1 and 146.3 dB.

⚠ QUIET TALKERS AT 8 BITS

A telephone call spans a wide range of levels. At 8 uniform bits a quiet talker keeps little SQNR. Companding, next, fixes this.

1.5 Non-uniform quantization

Non-uniform quantization

Speech is mostly quiet and only sometimes reaches its peak. A uniform step is the same for a whisper and a shout. So the whisper is quantized coarsely in proportion to itself.

Divide the bound $|q| \leq \Delta/2$ by $|m|$ to get the **relative error**. Small amplitudes suffer the most.

RELATIVE ERROR

$$\frac{|q|}{|m|} \leq \frac{\Delta/2}{|m|} = \frac{\Delta}{2|m|}$$

ⓘ COMPANDING

Compress the signal with a memoryless curve, quantize uniformly, and expand at the receiver. The name **companding** joins **com**pressing and ex**panding**.

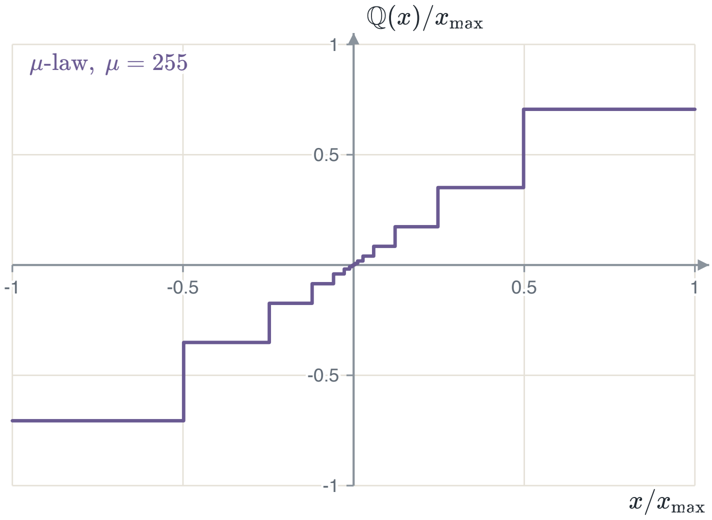


FIGURE 1.45 A sixteen-level μ-law quantizer. Its steps are fine near zero, where speech amplitudes crowd, and coarse near the peak.

For example, let a uniform quantizer have $\Delta = 0.1$ V and a quiet signal peak at 0.2 V. The error can reach $(\Delta/2)/0.2 = 0.05/0.2 = 25\%$ of that peak.

The compander

A **compander** builds a non-uniform quantizer from three parts: a compressor, a uniform quantizer and an expander.

> COMPRESSOR
>
> A memoryless curve $y = c(m)$. It gives a weak signal a high gain and a strong signal a low gain, so small amplitudes are spread over more quantizer levels.

> EXPANDER
>
> The receiver applies the inverse curve c^{-1}. It takes away the boost that the compressor gave the weak signal and returns every level to its original size.

The whole chain maps m to $\hat{m}$. Here Q is the uniform quantizer.

> COMPANDER
>
> $$\hat{m} = c^{-1}\big(Q(c(m))\big)$$
>
> Near an input m, a step Δ in y is a step of about $\Delta/c'(m)$ in m.

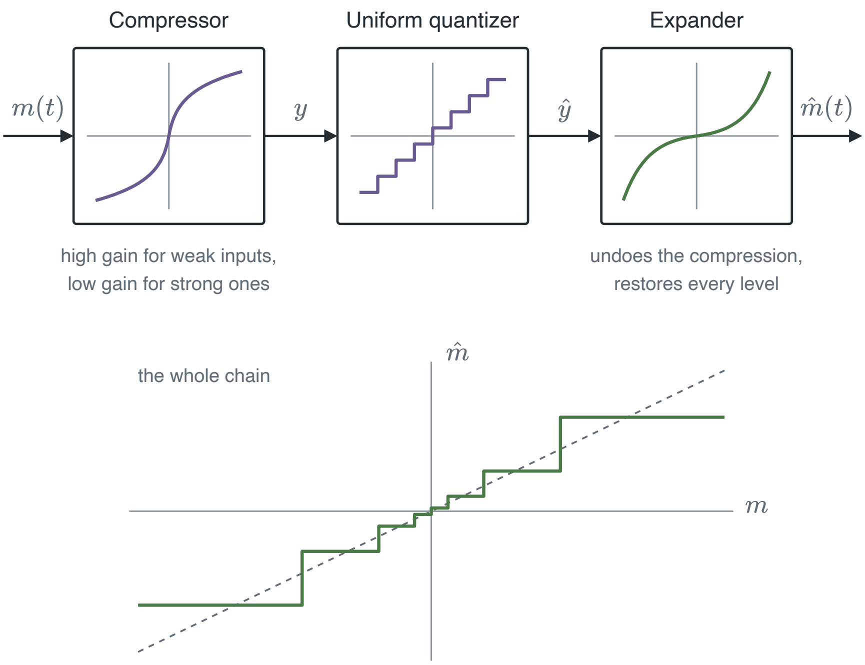


FIGURE 1.46 The quantizer in the middle is uniform. Seen from m to $\hat{m}$, its steps are fine near zero and coarse near the peak.

For example, let the compressor have slope $c'(0) = 4$ near zero and the uniform step be $\Delta = 0.1$. A quiet signal sees the step $\Delta/c'(0) = 0.1/4 = 0.025$. That step is four times finer than the uniform one.

A-law and μ-law companding

Two compressor curves are standard. Both act on an input normalized to $|x| \leq 1$. The first is the μ-law.

μ-LAW

$$y = \frac{\ln(1 + \mu|x|)}{\ln(1 + \mu)} \operatorname{sgn}(x), \qquad |x| \leq 1$$

The second is the A-law.

A-LAW

$$y = \begin{cases} \dfrac{A|x|}{1 + \ln A} \operatorname{sgn}(x), & |x| \leq \dfrac{1}{A} \\[2ex] \dfrac{1 + \ln(A|x|)}{1 + \ln A} \operatorname{sgn}(x), & \dfrac{1}{A} < |x| \leq 1 \end{cases}$$

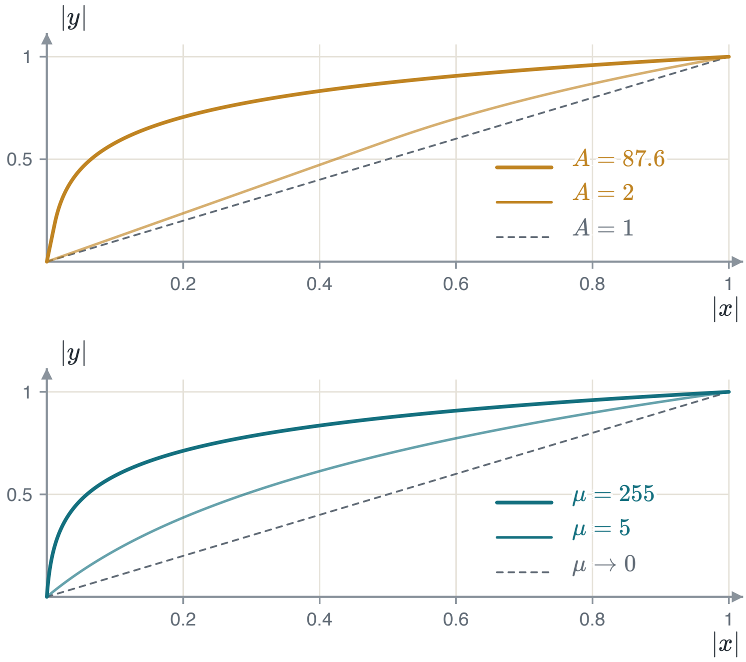


FIGURE 1.47 A-law (top) and μ-law (bottom) for three parameter values. The dashed line is no companding. The larger the parameter, the more output range goes to small inputs.

> IN USE
>
> μ-law with $\mu = 255$ is used in the United States, Canada and Japan. A-law with $A = 87.6$ is used in Europe, Türkiye and most other countries.

For example, with $\mu = 255$ the input $x = 0.01$ gives $y = \ln(3.55)/\ln(256) = 1.267/5.545 = 0.23$. One per cent of the range is lifted to almost a quarter.

Hearing companding

Compare a uniform quantizer and a μ-law quantizer, both with 6 bits over the same range. The uniform one spends its levels evenly, so a quiet voice meets only a few of them.

> QUIET VOICES STAY CLEAR
>
> The μ-law steps shrink with the voice. Its SQNR hardly changes as the level falls, while the uniform SQNR falls a decibel per decibel.

For example, let a talker drop 20 dB below full scale. The uniform noise power $\Delta^2/12$ stays the same and the signal power falls by 20 dB. So the uniform SQNR falls by 20 dB.

Companding around us

> TELEPHONE PCM
>
> North America and Japan use μ-law with $\mu = 255$. Most other countries use A-law with $A = 87.6$. Both use 8 bits at 8 kHz.

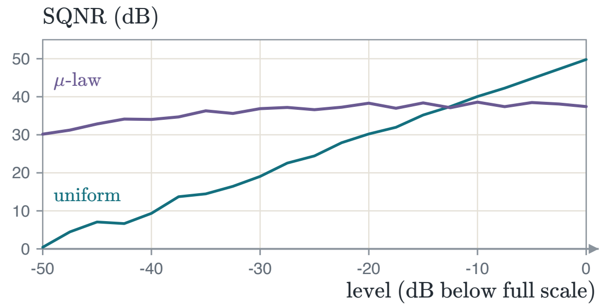


FIGURE 1.48 Two 8-bit quantizers on a sinusoid. The μ-law SQNR stays nearly flat as the level falls. The uniform one falls a decibel per decibel.

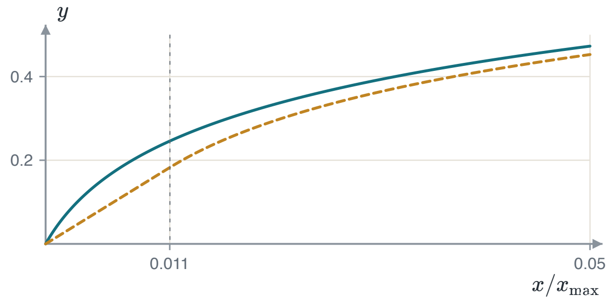


FIGURE 1.49 Near zero the laws differ: A-law is a straight line up to $1/A = 0.0114$, and μ-law is curved all the way.

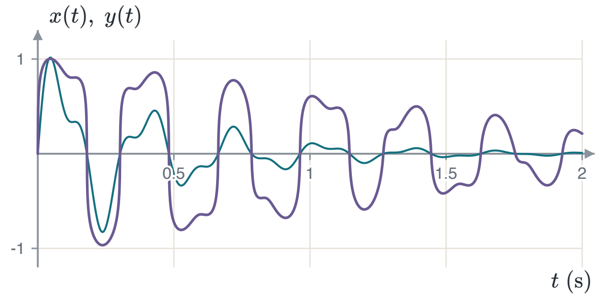


FIGURE 1.50 A decaying burst (cyan) and its μ-law compressed form (violet). The quiet tail is lifted toward full scale.

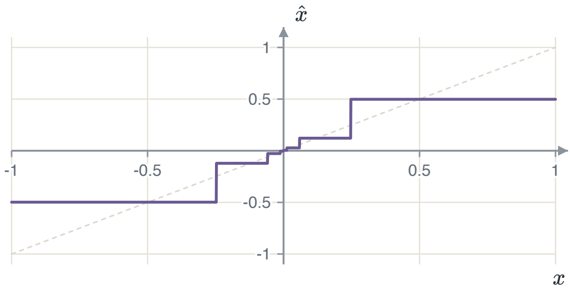


FIGURE 1.51 A 3-bit μ-law quantizer drawn against its input: narrow steps near zero, wide steps near the peak.

STEADY QUALITY

Companding gives up a little SQNR at full scale. In return the SQNR stays nearly constant over a wide range of input levels.

1.6 PCM, DPCM and delta modulation

Encoding and the bit rate

Pulse code modulation (PCM) writes each quantized sample as R bits. The bit rate is the bits a sample times the samples a second.

KEY RESULT: BIT RATE

$$R_b = R\, f_s \qquad \left(\frac{\text{bits}}{\text{sample}}\right)\left(\frac{\text{samples}}{\text{s}}\right)$$

TWO CODES

Natural binary numbers the levels 0 to $L-1$ in order. **Gray coding** orders the words so that adjacent levels differ in exactly one bit.

TABLE 1.3 The eight levels of a 3-bit quantizer in natural binary and in Gray code. The natural words 011 and 100 either side of the middle differ in all three bits.

LEVEL	NATURAL	GRAY
0	000	000
1	001	001
2	010	011
3	011	010
4	100	110
5	101	111
6	110	101
7	111	100

> GRAY-CODE ADVANTAGE
>
> A small decision error usually picks a neighbouring level. With a Gray code that costs one bit error, not several.

For example, telephone speech uses $L = 256$ levels at $f_s = 8$ kHz. Then $R = \log_2 256 = 8$ bits, and $R_b = 8(8000) = 64$ kb/s.

Line codes

> LINE CODE
>
> A **line code** is the rule that turns the bits of a PCM stream into a waveform on the wire.

Line codes differ in their levels. The levels decide whether the waveform has a DC component.

TABLE 1.4 The DC component of the two families of line code.

FAMILY	LEVELS	DC COMPONENT
Unipolar	0 and $+A$	present, and an AC-coupled stage cannot pass it
Polar	$\pm A$	none for balanced data

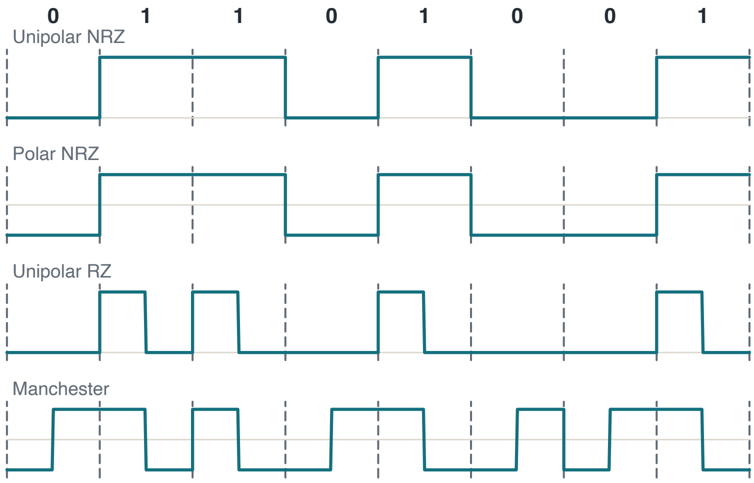


FIGURE 1.52 Four line codes of the bits $0\,1\,1\,0\,1\,0\,0\,1$. Each faint line is the zero of its trace.

> ⚠ CLOCK RECOVERY
>
> A long run of equal NRZ bits has no transitions. Manchester coding puts a transition in every bit, at the cost of twice the bandwidth.

For example, a polar NRZ stream that sends a long run of ones stays at $+A$. It has no transitions to lock onto, so the receiver cannot recover its clock from that stretch.

EXAMPLE 1.5 — PCM ENCODING OF A SINC PULSE

GIVEN $m(t) = 8\,|\operatorname{sinc}(t-2)|$, with $\operatorname{sinc}(x) = \sin(\pi x)/(\pi x)$. It is sampled every $T_s = 0.6$ s and quantized by an eight-level uniform quantizer over $[0, 8]$.

FIND The step size, the code words for $t = 0, 0.6, \ldots, 3.6$, and the bit rate.

METHOD Take Δ from the range and L. Evaluate each sample. Read the tread it falls in, then its code word. The bit rate is $R_b = R\,f_s$.

SOLUTION The step and the levels come from the range $[0, 8]$ and $L = 8$.

$$\begin{aligned}\Delta &= \frac{8-0}{8} = 1\ \mathrm{V}\\ v_k &= \left(k + \tfrac{1}{2}\right)\Delta, \quad k = 0, \ldots, 7\\ &= 0.5,\ 1.5,\ \ldots,\ 7.5\ \mathrm{V}\end{aligned}$$

Evaluate the samples. For instance, $m(0.6) = 8\,|\sin(1.4\pi)|/(1.4\pi) = 8(0.951)/4.398 = 1.73$.

t	0	0.6	1.2	1.8	2.4	3.0	3.6
$t-2$	-2	-1.4	-0.8	-0.2	0.4	1	1.6
$m(t)$	0	1.73	1.87	7.48	6.05	0	1.51

Each sample takes the level of its tread. The index is $k = v/\Delta - \frac{1}{2}$, written in three bits.

v	0.5	1.5	1.5	7.5	6.5	0.5	1.5
k	0	1	1	7	6	0	1
code	000	001	001	111	110	000	001

Three bits a sample give the bit rate and the bit duration.

$$f_s = \frac{1}{T_s} = \frac{1}{0.6} = 1.667\ \mathrm{samples/s}$$
$$R_b = R\,f_s = \frac{3}{0.6} = 5\ \mathrm{b/s}$$
$$T_b = \frac{1}{R_b} = 0.2\ \mathrm{s}$$

CHECK At $t = 3$ the sample is $8|\,\mathrm{sinc}(1)| = 0$ exactly, because sinc vanishes at every non-zero integer.

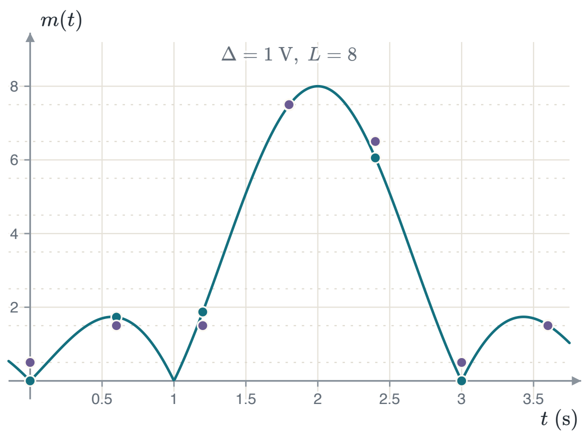


FIGURE 1.53 The message (cyan), its samples every 0.6 s, and the selected levels (violet). Each error is under half a step.

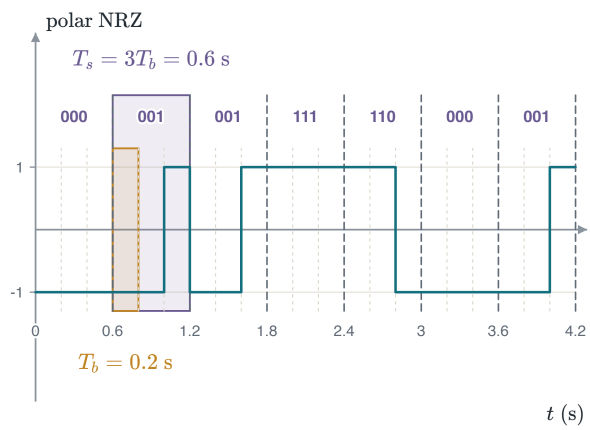


FIGURE 1.54 The seven code words as one polar NRZ stream. The amber box marks one bit, $T_b = 0.2$ s. The violet box marks one code word, $T_s = 3T_b$.

The bandwidth of PCM

A binary stream of R_b bits a second needs a bandwidth of at least $R_b/2$ hertz. Chapter 2 shows why. At the Nyquist rate $f_s = 2W$ this becomes RW.

KEY RESULT: BANDWIDTH OF PCM

$$B_T \geq \frac{R_b}{2} = \frac{R\,f_s}{2}$$
$$B_T \geq RW \quad \text{at } f_s = 2W$$

BITS FOR BANDWIDTH

Each extra bit adds 6.02 dB of SQNR and W hertz of bandwidth. PCM buys quality with bandwidth.

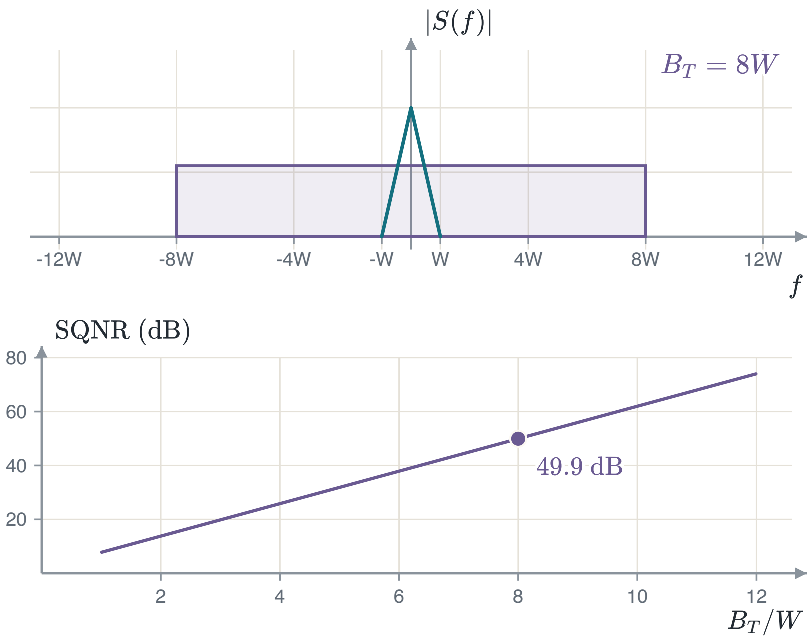


FIGURE 1.55 Top: the message band W (cyan) and the least band of 8-bit PCM sampled at $2W$ (violet). Bottom: the SQNR of a full-scale sinusoid against that bandwidth.

For example, telephone PCM with $f_s = 8$ kHz and $R = 8$ bits has $R_b = 8(8000) = 64$ kb/s. Its least bandwidth is $B_T \geq R_b/2 = 32$ kHz, eight times the 4 kHz of the voice.

Bit errors in PCM

A channel error flips one bit of a code word. With natural binary coding the damage depends on the position of the bit. Bit b counts from 0 at the right and carries the weight 2^b.

> SIZE OF THE ERROR
>
> $$\hat{m} - m_q = \pm 2^b \Delta$$
>
> Here m_q is the level sent and $\hat{m}$ the level decoded. The first bit, $b = R - 1$, moves the sample by $2^{R-1}\Delta = m_{\max}$, half the range.

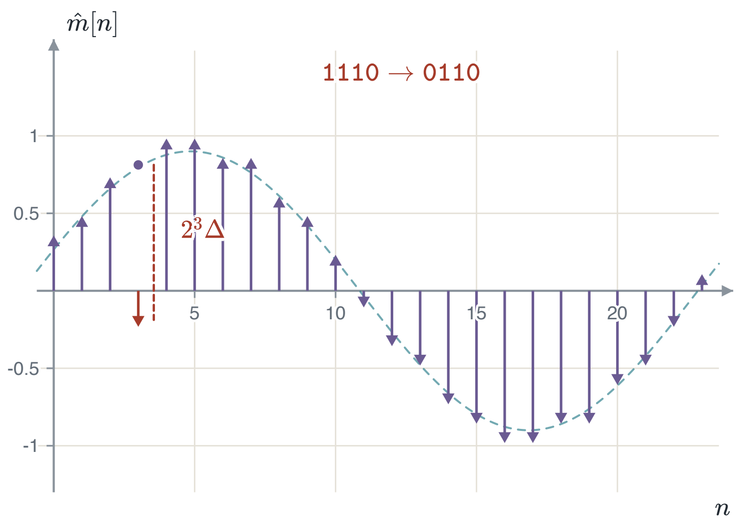


FIGURE 1.56 A 4-bit PCM stream with $\Delta = 0.125$. Bit $b = 3$ of the code word 1110 is received wrong, and the sample moves by $2^3\Delta$.

⚠ CLICKS

A wrong first bit throws a sample across half the range. The ear hears a click, far louder than the quantization noise.

For example, an 8-bit quantizer with $\Delta = 1/128$ V moves a sample by $2^7\Delta = 128/128 = 1$ V when the first bit is wrong. That is $m_{\max}$.

Differential PCM

ⓘ PREDICT, THEN SEND THE DIFFERENCE

Neighbouring samples are close. **Differential PCM** (DPCM) quantizes the difference between a sample and its prediction, which has a much smaller range.

The prediction is the last decoded value $\hat{x}[n-1]$. The encoder forms the difference, and the decoder adds the quantized difference to its last value.

ENCODER AND DECODER

$$e[n] = x[n] - \hat{x}[n-1]$$
$$\hat{x}[n] = \hat{x}[n-1] + \mathbb{Q}\big(e[n]\big)$$

The encoder predicts from $\hat{x}[n-1]$, the value the decoder also has. So $x[n] - \hat{x}[n] = e[n] - \mathbb{Q}(e[n])$, and errors do not pile up.

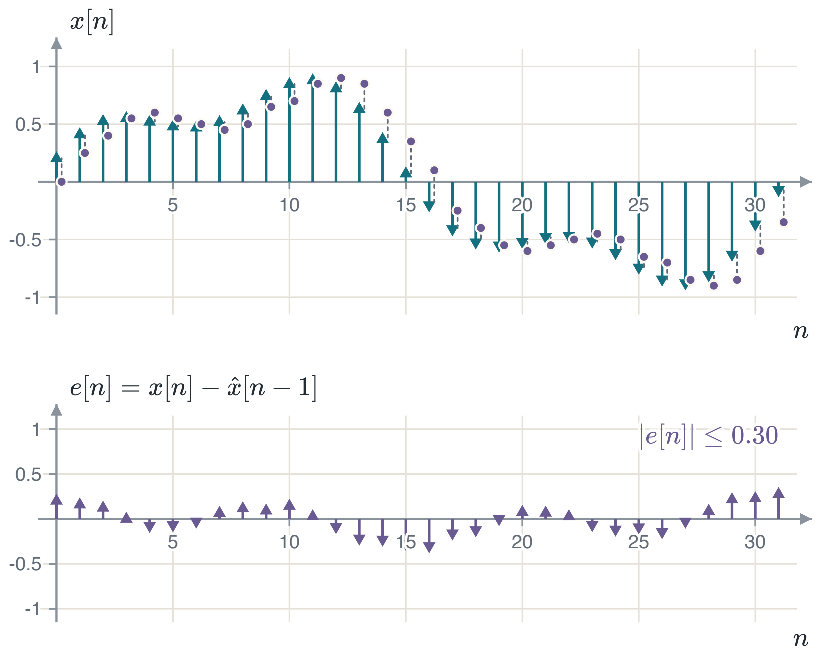


FIGURE 1.57 Each sample is predicted by the last decoded value (violet dots). The difference, on the same scale below, is far smaller than the sample.

FEWER BITS

For speech at 8 kHz, DPCM with 4 bits a sample matches PCM with 8. That is 32 kb/s in place of 64 kb/s.

For example, DPCM with 4 bits a sample at $f_s = 8$ kHz has $R_b = R\,f_s = 4(8000) = 32$ kb/s.

Delta modulation

ONE BIT A SAMPLE

Delta modulation is DPCM with a two-level quantizer. Each bit moves the staircase up or down by Δ.

The step size trades two errors against each other.

TABLE 1.5 The two errors of delta modulation.

ERROR	CAUSE
Slope overload	The signal rises faster than Δ a sample. The staircase falls behind.
Granular noise	The signal is flat. The staircase hunts by $\pm\Delta$ around it.

The staircase climbs at most Δ every T_s. It follows the signal only while the slope stays below that climb.

NO OVERLOAD

$$\left|\frac{dx}{dt}\right| \le \frac{\Delta}{T_s} = \Delta f_s$$

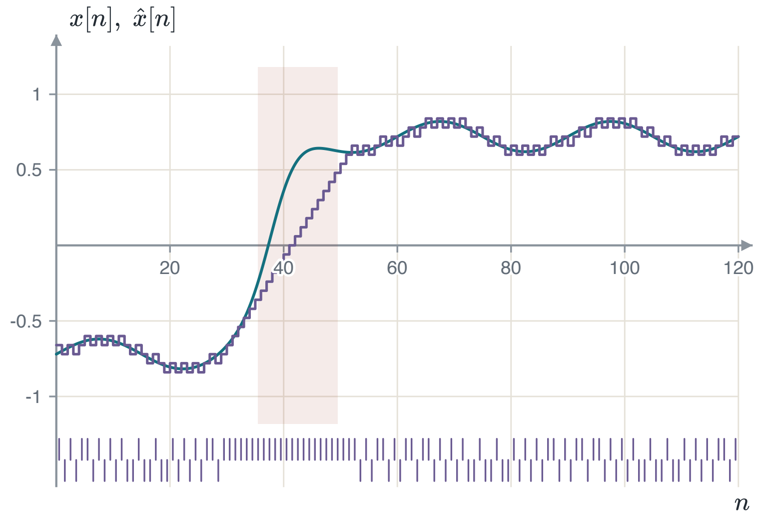


FIGURE 1.58 A signal with one steep rise, and the staircase with $\Delta = 0.06$ that follows it one step a sample. Red shades slope overload. The ticks under the plot are the bits.

For example, $x(t) = \sin(2\pi\,1000\,t)$ has largest slope $2\pi(1000)$, and the rate is $f_s = 64$ kHz. No overload needs $\Delta \ge 2\pi(1000)/64\,000 = 0.098$.

PCM around us

> **RATE FROM THREE NUMBERS**
>
> The bit rate is the bits a sample, times the sampling rate, times the number of channels.

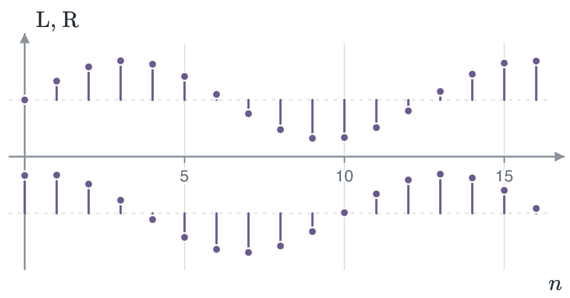


FIGURE 1.59 CD audio: two channels of 16 bits at 44.1 kHz give $2(16)(44\,100) = 1.4112$ Mb/s.

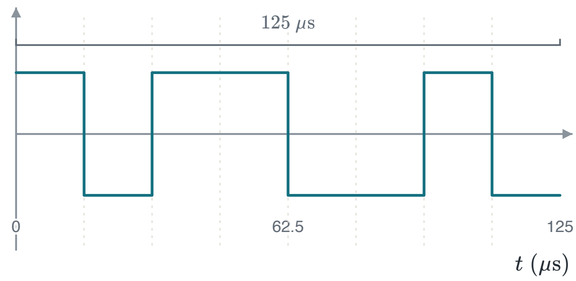


FIGURE 1.60 A telephone call sends one 8-bit word every 125 μs, which is 64 kb/s.

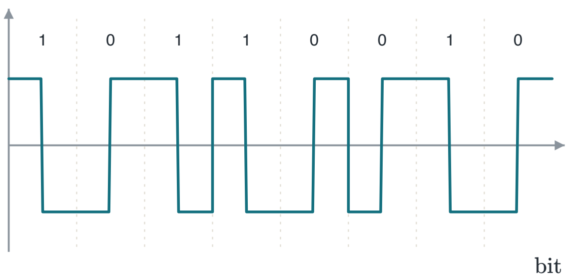


FIGURE 1.61 Ten-megabit Ethernet over twisted pair uses Manchester coding: a transition in the middle of every bit.

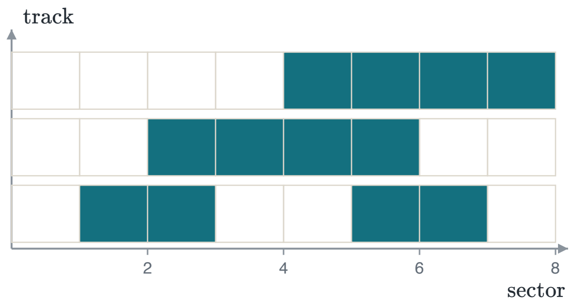


FIGURE 1.62 A shaft encoder prints a Gray code on three tracks. At each sector edge only one track changes.

> **ONE BIT AT A BOUNDARY**
>
> A reading taken exactly on a sector edge can see either neighbour. With a Gray code it is off by at most one sector.

1.7 Vector quantization

Vector quantization

> **VECTOR QUANTIZATION**
>
> **Vector quantization** treats n samples as one point in n dimensions and sends the nearest point of a **codebook**. Scalar quantization is the case $n = 1$.

Take pairs of samples from a 16-level quantizer. Quantized one at a time, every pair of levels must be nameable.

EVERY PAIR

$$L^2 = 16^2 = 256 \text{ pairs}$$
$$\log_2 256 = 8 \text{ bits a pair}$$

Now let neighbouring samples differ by at most one step. Only pairs of levels with $|i - j| \leq 1$ occur. The diagonal holds L such pairs and each side line $L - 1$.

PAIRS THAT OCCUR

$$\begin{aligned} L + 2(L-1) &= 3L - 2 \\ &= 3(16) - 2 = 46 \text{ pairs} \\ \lceil \log_2 46 \rceil &= \lceil 5.52 \rceil = 6 \text{ bits a pair} \end{aligned}$$

Six bits a pair is three bits a sample, not four.

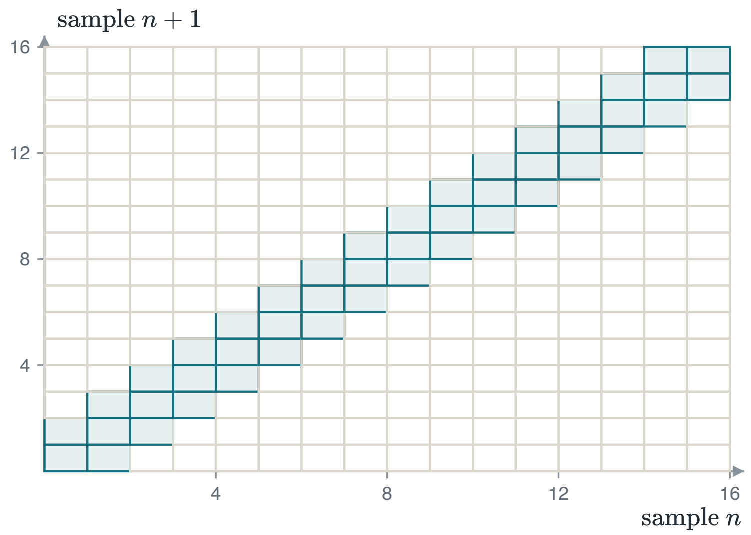


FIGURE 1.63 All 256 pairs of a 16-level quantizer. If neighbours differ by at most one step, only the 46 shaded pairs occur.

For example, an 8-level quantizer whose neighbours differ by at most one step has $3L - 2 = 3(8) - 2 = 22$ pairs that can occur.

Image quantization

A greyscale image is quantized pixel by pixel. The size of the file is the pixel count times the bits a pixel. One KiB is $8(1024)$ bits, and 32 levels need $\log_2 32 = 5$ bits.

BITS IN A 512×512 IMAGE

$$\begin{aligned} 512^2(8) &= 2\,097\,152 \text{ bits} \qquad & 512^2(5) &= 1\,310\,720 \text{ bits} \\ &= 256 \text{ KiB} & &= 160 \text{ KiB} \end{aligned}$$

The file shrinks by $(256 - 160)/256 = 37.5\%$.

Three bits fewer cost three times six decibels of SQNR.

WHAT IT COSTS

$$6.02(8-5) = 6.02(3) = 18.06 \text{ dB}$$

FIGURE 1.64 A grey ramp from 0 to 1 through a uniform quantizer with $L = 8$ levels. Below it is the size of a 512×512 image at 3 bits a pixel.

⚠ BANDING

Coarse levels turn a smooth gradient into flat steps. The eye reads the step edges as contours that the scene never had.

For example, a 256×256 image with $L = 64$ levels needs $R = \log_2 64 = 6$ bits a pixel. Its size is $256^2(6)/8 = 49\,152$ bytes, or 48 KiB.

1.8 Speech, audio and image coding

Linear predictive coding

ⓘ A MODEL, NOT A WAVEFORM

Linear predictive coding (LPC) measures a model of the voice every 20 ms and sends its settings. The receiver runs the model to make the speech again.

The model is an all-pole filter driven by an excitation w_n. Each output sample is a weighted sum of the last p outputs plus the scaled excitation.

ALL-POLE MODEL

$$x_n = \sum_{i=1}^{p} a_i\, x_{n-i} + G\, w_n$$

w_n is a pulse train for a voiced sound and white noise for an unvoiced one. The a_i describe the shape of the vocal tract.

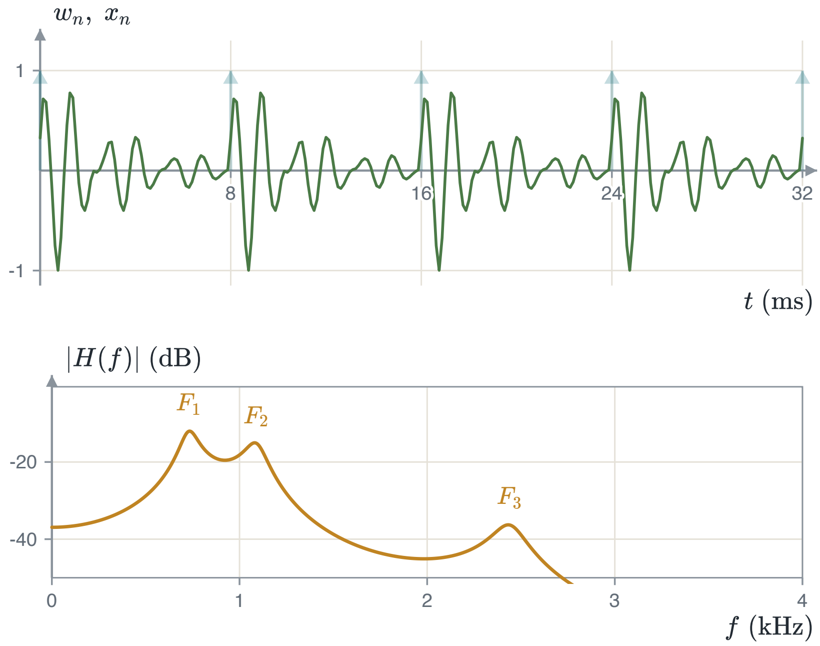


FIGURE 1.65 A pulse train at $f_0 = 125$ Hz (faint) drives an all-pole filter with three peaks, the formants F_1, F_2, F_3. The output is a vowel (green). Noise through the same filter is a whisper.

> **BIT RATE**
>
> Each frame sends the a_i, the gain G, the pitch and one voiced bit. That brings speech down to about 2.4 kb/s, against 64 kb/s for PCM.

For example, a coder that sends 48 bits for each 20 ms frame sends 50 frames a second. Its bit rate is $48(50) = 2400$ b/s.

Time-division multiplexing and T1

> **TIME-DIVISION MULTIPLEXING**
>
> Many PCM calls share one line by taking turns. Each call gets one slot in every frame, and a frame lasts one sampling interval.

The T1 line carries 24 calls of 8 bits each, plus one framing bit. The extra bit marks the start of each frame, so the receiver knows which slot is which. A frame is sent every $125\ \mu$s, that is 8000 times a second.

> **T1 LINE RATE**
>
> $$24(8) + 1 = 193 \text{ bits a frame}$$
> $$193(8000) = 1.544 \text{ Mb/s}$$

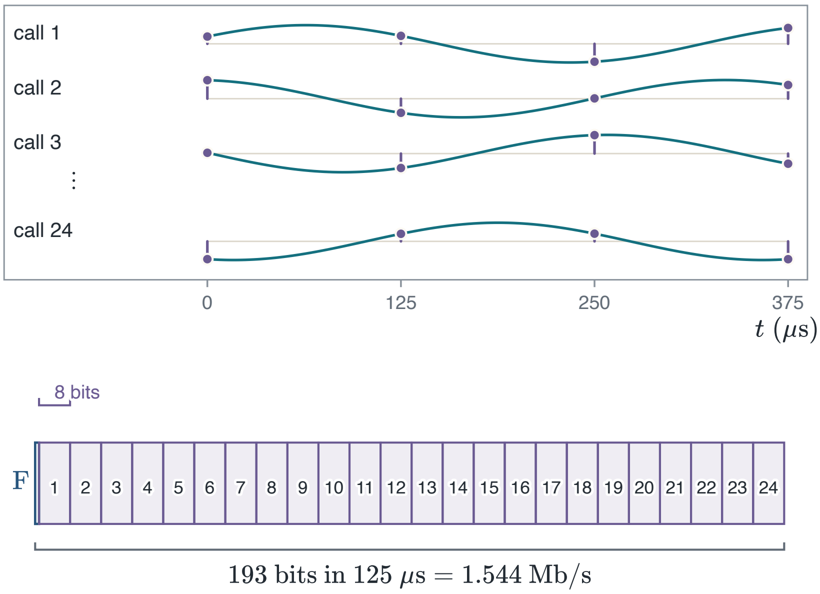


FIGURE 1.66 Every call is sampled every 125 μs. Its 8-bit word takes one slot of the frame, after a framing bit F.

> A HIERARCHY
>
> Four T1 lines make a 6.312 Mb/s line, and seven of those a 44.736 Mb/s line. Europe and Türkiye use E1: 32 slots, 2.048 Mb/s.

For example, leave out the framing bit. The 24 calls alone need $24(64) = 1536$ kb/s. The framing bit adds 8 kb/s to make 1.544 Mb/s.

Oversampling and sigma-delta conversion

> OVERSAMPLE, THEN USE ONE BIT
>
> At a rate far above $2W$, neighbouring samples are nearly equal. One bit a sample can then carry the change.

A **sigma-delta** converter integrates the error between the input and its one-bit output. The output bit is the sign of the integrator.

> SIGMA-DELTA LOOP
>
> $$v[n] = v[n-1] + x[n] - y[n-1]$$
> $$y[n] = \operatorname{sgn}\big(v[n]\big)$$
>
> The integrator v adds up the error, so the running average of the bits y stays on the input.

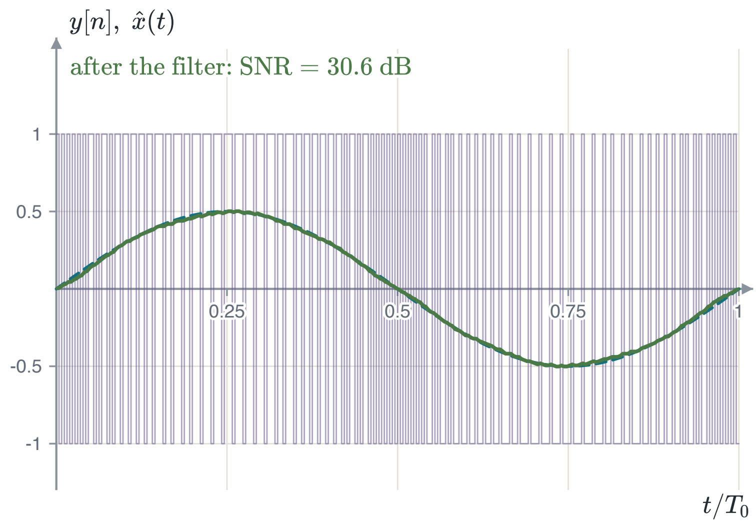


FIGURE 1.67 One period of a sinusoid (dashed) oversampled $U = 16$ times. The one-bit output (violet) swings between ± 1, and a lowpass filter averages it back to the signal (green).

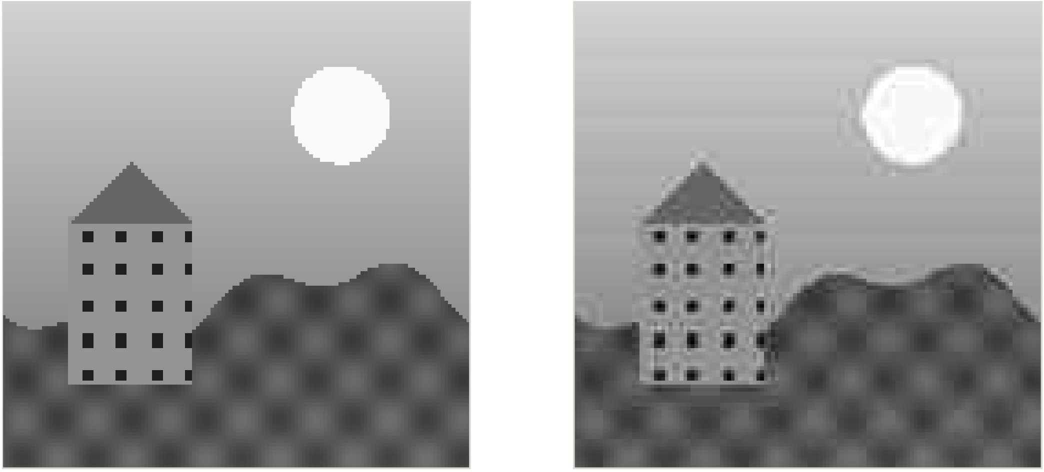
> **THE DECODER IS A FILTER**
> A lowpass filter turns the bits back into the signal. The fast swings of the bits lie far above 20 kHz and are removed.

For example, a CD player that oversamples 44.1 kHz audio by $U = 256$ runs its one-bit converter at $256(44.1\ \text{kHz}) = 11.2896$ MHz.

Transform coding and JPEG

> **TRANSFORM CODING**
> JPEG cuts the picture into 8×8 blocks and takes the discrete cosine transform of each. Most of the energy of a block lands in a few low-frequency coefficients.

> **QUANTIZE THE COEFFICIENTS**
> Each coefficient has its own uniform step, larger at high frequencies. Most high-frequency coefficients round to zero and cost almost nothing to send.



original

quality 25

1383 of 16384 coefficients are not zero (8.4%)

FIGURE 1.68 A drawn picture of 128×128 pixels and its JPEG version at quality 25. At lower quality more coefficients become zero, and the 8×8 blocks begin to show.

> ⚠ BLOCKING
>
> At a coarse step each block keeps little more than its average. The picture breaks into visible 8×8 squares.

For example, a 512×512 picture coded at 0.5 bit a pixel takes $512^2(0.5)/8 = 16\,384$ bytes, or 16 KiB. That is sixteen times less than at 8 bits a pixel.

1.9 Summary

From sound to bits

One signal passes through the whole chain in four stages.

1. Filter the signal to the band W.
2. Sample at $f_s \geq 2W$.
3. Quantize to $L = 2^R$ levels.
4. Write R bits a sample.

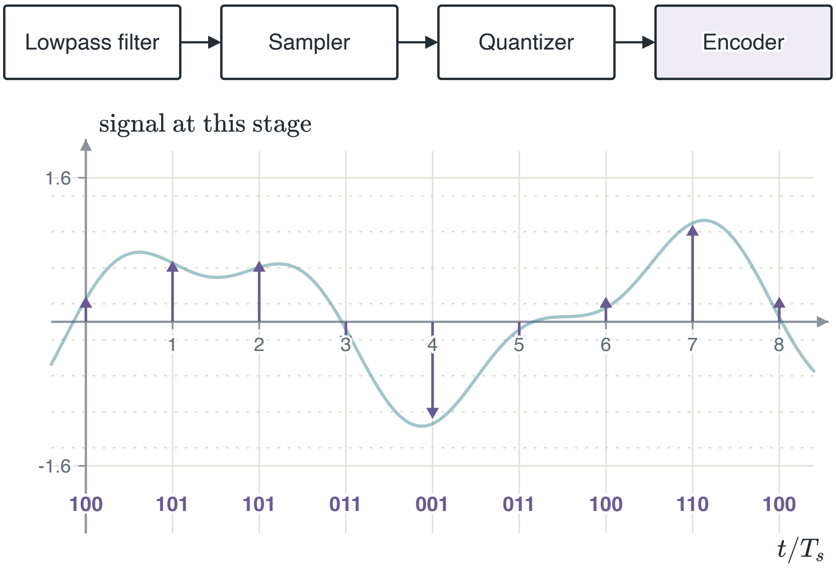


FIGURE 1.69 One message through the four stages. The filter has removed a fast ripple, and the sampler has kept one value every T_s. The quantizer has moved each value to one of eight levels, and the encoder has written three bits for it.

> QUANTIZATION ERROR
>
> Filtering and sampling at $f_s \geq 2W$ keep everything inside the band. Only the quantizer adds an error that no receiver can remove.

For example, a recording sampled at 16 kHz with 12 bits a sample has $R_b = R\,f_s = 12(16\,000) = 192$ kb/s.

Quick check

Each question needs no calculation on paper. The answer and its reason follow the question.

1 Aliasing. A 7 kHz tone sampled at 10 kHz appears at what frequency?
Answer: 3 kHz. Since $7 > f_s/2 = 5$, it folds to $|7 - 10| = 3$ kHz.

2 Nyquist rate. What is the Nyquist rate of audio bandlimited to 20 kHz?
Answer: $2W = 40$ kHz. CD audio samples a little above it.

3 The sinc convention. What is $\mathrm{sinc}(2)$?
Answer: 0. Since $\sin(2\pi)/(2\pi) = 0$, sinc vanishes at every non-zero integer.

4 Bits and SQNR. By about how much do two more bits a sample raise the SQNR?
Answer: 12 dB. At 6.02 dB a bit, two bits give 12.04 dB.

5 Step size. Doubling Δ multiplies $E[Q^2]$ by what factor?
Answer: 4. $E[Q^2] = \Delta^2/12$ grows with the square of the step.

6 Companding. Does μ-law companding give finer steps to small or to large amplitudes?
Answer: Small amplitudes. The compressor is steep near zero, so small inputs spread over many levels.

7 Gray code. In how many bits do two adjacent Gray code words differ?
Answer: One bit. That is the rule that defines the code.

8 Levels. How many levels does a 16-bit quantizer have?
Answer: $L = 2^{16} = 65\,536$.

Summary of results

TABLE 1.6 Summary of Chapter 1: from an analog signal to bits.

QUESTION	RESULT	ANCHOR
What does sampling do to the spectrum?	$G_\delta(f) = f_s \sum_n G(f - nf_s)$: a copy at every multiple of f_s, scaled by f_s.	PS CH7.1.1
What is the lowest safe sampling rate?	The Nyquist rate $2W$. Below it the copies overlap and alias.	PS CH7.1.1
What filter turns the samples back into the signal?	An ideal lowpass filter with gain T_s. At $f_s = 2W$ its impulse response is $\operatorname{sinc}(2Wt)$.	PS CH7.1.1
How is $g(t)$ written in terms of its samples?	$g(t) = \sum_n g(nT_s) \operatorname{sinc}\big(2W(t - nT_s)\big)$, exact at every t.	PS CH7.1.1
Mid-rise or mid-tread: which has a level at zero?	Mid-tread. Mid-rise has a boundary at zero and outputs $\pm\Delta/2$ there.	PS CH7.2.1
What is the power of the quantization noise?	$E[Q^2] = \Delta^2/12$, with $\Delta = 2m_{\max}/L$, for a fine quantizer.	PS CH7.2.1
What does one more bit buy?	6.02 dB: $\mathrm{SQNR} = \alpha + 6.02R$, and $\alpha = 1.76$ dB for a full-scale sinusoid.	PS CH7.2.1
Why compand?	Small amplitudes are common. Finer steps near zero keep the SQNR steady as the level falls.	PS CH7.2.1
What bit rate does PCM need?	$R_b = R\,f_s$. Telephone speech: $8(8000) = 64$ kb/s.	PS CH7.3
Why use a Gray code?	Adjacent levels differ in one bit, so a small decision error costs one bit error.	PS CH7.3
What bandwidth does PCM need?	$B_T \geq R_b/2$, which is RW at $f_s = 2W$. Each extra bit adds 6.02 dB and W hertz.	PS CH7.4.1
What do DPCM and delta modulation send?	The difference from a prediction. Delta modulation sends one bit a sample and needs Δf_s above the largest slope.	PS CH7.4.2–7.4.3

> **METHOD**
> Check a rate against $2W$ before sampling. Take Δ from the full range $2m_{\max}$. Compute the SQNR from the two powers and compare it with $\alpha + 6.02R$. Chapter 2 sends the resulting bits over a channel.

Projects to try

Four optional projects use the chapter on real and computed signals. Each gives an aim, what it practises, a few steps and what to look for.

HEAR ALIASING IN A RECORDING

Lower the sampling rate of a recording and hear what aliasing does to it.
Practises: The Nyquist rate $2W$ of a real signal. Aliasing: a tone above $f_s/2$ returns at a lower frequency. Why a lowpass filter comes before the sampler.

1. Record a few seconds of music or speech at $f_s = 44.1$ kHz.
2. Keep every 4th sample and play the result at $f_s/4$. Then keep every 8th sample.
3. Repeat, but first remove everything above the new $f_s/2$ with a lowpass filter.
4. Do the same with a tone that sweeps from 0 to 10 kHz, and plot the spectrum of each version.

Look for: Without the filter, high notes come back as new low notes that were never played. With the filter, the sound is only duller. The swept tone climbs, turns and falls, again and again.

REBUILD A TONE FROM ITS SAMPLES

Rebuild a signal from its samples in two ways and measure how close each one gets.
Practises: The interpolation formula $g(t) = \sum_n g(nT_s)\,\mathrm{sinc}\big(2W(t - nT_s)\big)$, with $\mathrm{sinc}(x) = \sin(\pi x)/(\pi x)$. The zero-order hold, which keeps each sample until the next one. The error of a finite sum of sinc terms.

1. Sample $g(t) = \cos(2\pi\, 3t)$ at $f_s = 8$ Hz on $0 \le t \le 4$ s.
2. Rebuild $g(t)$ on a fine grid with the sinc sum. Rebuild it again with a zero-order hold.
3. Plot both against the true $g(t)$, and plot the error of each.
4. Repeat with $f_s = 6.5$ Hz and with $f_s = 5$ Hz.

Look for: The sinc sum is close in the middle and worse near the two ends, where terms are missing. The hold lags by half a sample. At $f_s = 5$ Hz both rebuild a different tone.

TAKE BITS AWAY FROM A SONG

Quantize a recording with fewer and fewer bits and compare the measured SQNR with the rule.
Practises: A uniform quantizer with $L = 2^R$ levels and $\Delta = 2m_{\max}/L$. The SQNR measured from the signal power and the error power. When the model of the error as uniform noise fails.

1. Scale a recording so that $|x| \le m_{\max}$. Quantize it with $R = 12, 8, 6, 4, 2$ bits.
2. For each R, compute the error $q = x_q - x$ and the SQNR $= 10\log_{10}\big(\overline{x^2}/\overline{q^2}\big)$.
3. Plot SQNR against R and draw the line $\alpha + 6.02R$ over it.
4. Listen to x_q and to q alone for each R.

Look for: The points follow a straight line for large R and leave it for small R. There the error sounds like the music, not like a hiss. α is lower than for a full-scale sinusoid.

KEEP A QUIET VOICE CLEAR

Compare a uniform quantizer with a μ-law quantizer as the input level falls.

Practises: The μ-law compressor and its inverse, the expander. SQNR against input level for both quantizers. Why speech and telephony use companding.

1. Record a sentence and scale its peak to $m_{\max}$. Make copies scaled down by 10, 20, 30 and 40 dB.
2. Quantize each copy with 8 bits: once uniformly, once through the compressor with $\mu = 255$, then the expander.
3. Compute the SQNR of every version and plot it against the input level.
4. Listen to the quietest copy in both versions.

Look for: The uniform curve falls by about the same number of dB as the input. The μ-law curve stays nearly flat, so the quiet copy keeps most of its SQNR.

CHAPTER

2

Baseband transmission of digital signals

Each bit is sent as one waveform. The channel adds noise, and the receiver filters, samples and decides.

The matched filter gives the largest peak SNR, $2E/N_0$. Only the pulse energy counts, not its shape. With polar signalling and equal priors, $P_b = Q\left(\sqrt{2E_b/N_0}\right)$. A channel narrower than $R_b/2$ cannot avoid intersymbol interference.

2.1 The matched filter

The receiver model

One bit is sent as a pulse $g(t)$ on $0 \le t \le T$. The channel adds noise $w(t)$. The receiver passes the sum through a linear filter $h(t)$ and samples the output once, at $t = T$.

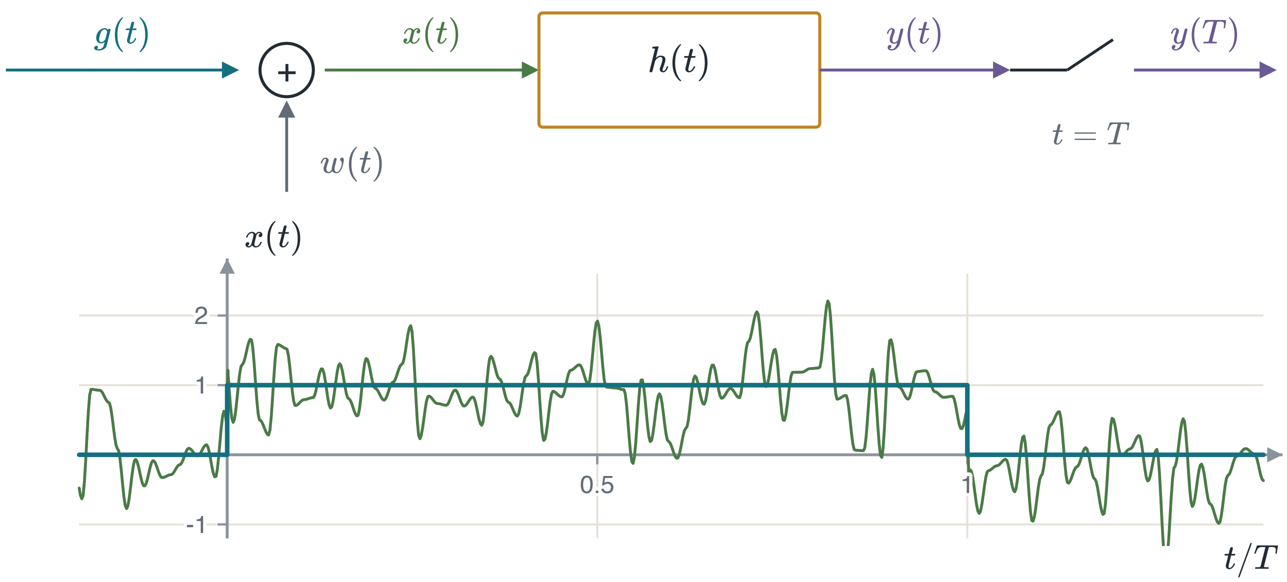


FIGURE 2.1 The receiver model. One pulse $g(t)$ on $[0, T]$ plus white noise $w(t)$ enters a linear filter, and the output is sampled once, at $t = T$. The lower panel shows one pulse (cyan) and the same pulse with noise on it (green).

RECEIVED SIGNAL

$$x(t) = g(t) + w(t), \qquad 0 \le t \le T$$

$g(t)$ is the pulse of one bit. $w(t)$ is the noise the channel adds.

The noise is white and Gaussian. White means that its power spectral density is flat. Its autocorrelation is then an impulse, so noise values at two different times are uncorrelated.

WHITE NOISE

$$S_W(f) = \frac{N_0}{2} \quad \text{for all } f, \qquad R_W(\tau) = \frac{N_0}{2}\,\delta(\tau)$$

The density is two-sided: it is $N_0/2$ at every positive and every negative frequency. For example, $S_W(f) = 10^{-9}$ W/Hz means $N_0 = 2(10^{-9}) = 2 \times 10^{-9}$ W/Hz.

The peak pulse signal-to-noise ratio

The filter is linear, so its output splits into two parts. The signal part $g_0(t)$ is the output for $g(t)$ alone. The noise part $n(t)$ is the output for $w(t)$ alone.

The receiver reads one value, at $t = T$. The quality of that value is the signal power there against the noise power there.

PEAK PULSE SNR

$$\eta = \frac{|g_0(T)|^2}{E[n^2(T)]}$$

The ratio η is the **peak pulse signal-to-noise ratio**. For example, $g_0(T) = 2$ mV and $E[n^2(T)] = 10^{-6}\ \mathrm{V}^2$ give $\eta = (2 \times 10^{-3})^2/10^{-6} = 4$.

Both parts can be written in the frequency domain. The signal part is the inverse transform of $H(f)G(f)$, read at $t = T$. The noise part integrates the output noise density $S_W(f)|H(f)|^2$.

BOTH PARTS IN f

$$g_0(T) = \int_{-\infty}^{\infty} H(f)\,G(f)\,e^{j2\pi fT}\,df$$
$$E[n^2(T)] = \int_{-\infty}^{\infty} \frac{N_0}{2}\,|H(f)|^2\,df$$

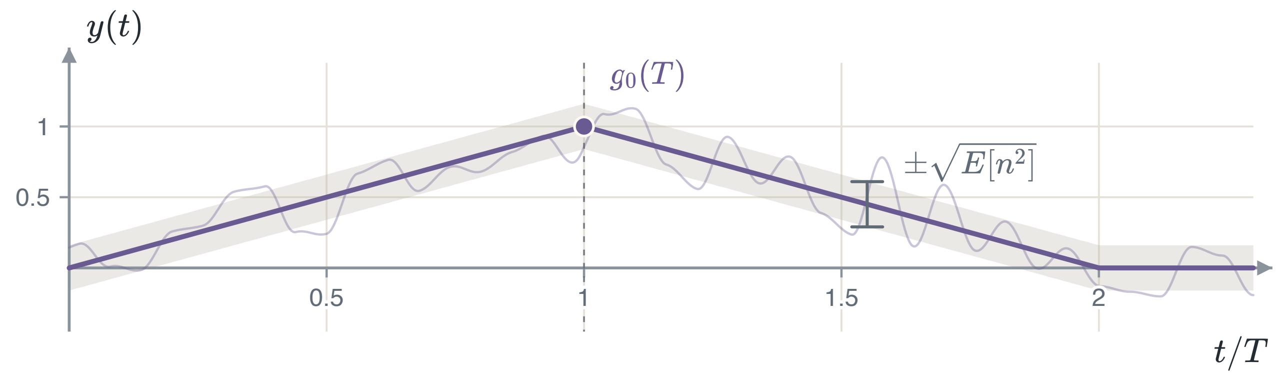


FIGURE 2.2 The filter output. The heavy line is the signal part $g_0(t)$ and the grey band is one noise standard deviation wide around it. The thin line is one noisy output. The receiver reads one value, at $t = T$.

The bound on the output SNR

Schwarz's inequality bounds η. It holds for any pair of functions.

> SCHWARZ'S INEQUALITY
>
> $$\left|\int \phi_1(f)\,\phi_2(f)\,df\right|^2 \le \int |\phi_1(f)|^2\,df \int |\phi_2(f)|^2\,df$$
>
> Equality holds only when $\phi_1(f) = k\,\phi_2^*(f)$ for a constant k.

Take $\phi_1 = H(f)$ and $\phi_2 = G(f)\,e^{j2\pi fT}$. The left side is then $|g_0(T)|^2$. Since $|e^{j2\pi fT}| = 1$, the right side is $\int |H(f)|^2 df \int |G(f)|^2 df$.

$$|g_0(T)|^2 \le \int |H(f)|^2 df \int |G(f)|^2 df$$

Divide both sides by the noise power $E[n^2(T)]$.

> THE BOUND
>
> $$\eta \le \frac{\int |H(f)|^2 df \int |G(f)|^2 df}{\frac{N_0}{2}\int |H(f)|^2 df} = \frac{2}{N_0}\int |G(f)|^2\,df = \frac{2E}{N_0}$$
>
> The factor $\int |H(f)|^2 df$ cancels between the top and the bottom. Parseval's theorem gives $\int |G(f)|^2 df = E$, the pulse energy, so the bound contains no H. For example, $E = 5\times10^{-7}$ J and $N_0 = 10^{-7}$ W/Hz give $\eta_{\max} = 2(5\times10^{-7})/10^{-7} = 10$.

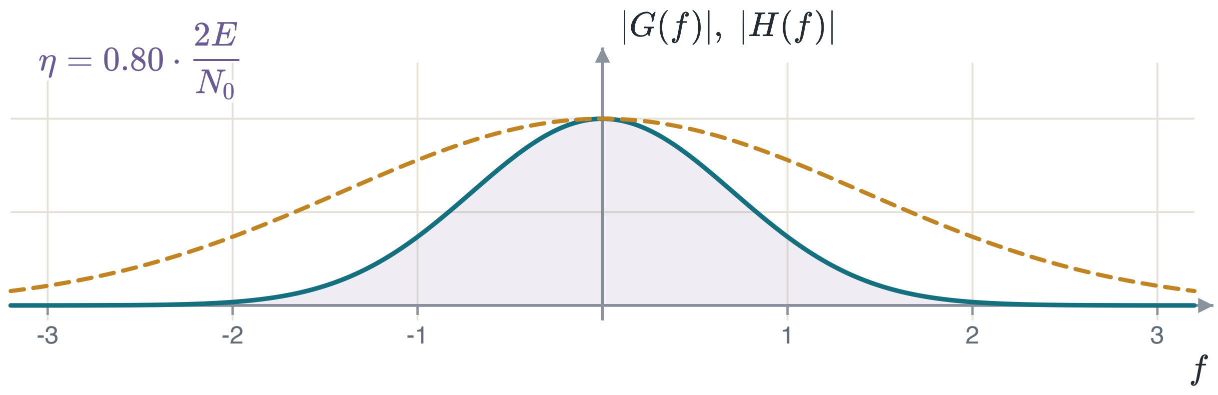


FIGURE 2.3 The signal spectrum $|G(f)|$ (cyan) and a filter $|H(f)|$ (amber, dashed) of twice its width, $b = 2$. The shaded overlap sets the SNR, here 0.80 of the bound. The bound is reached only when the two shapes agree, at $b = 1$.

The matched filter

The equality condition of Schwarz's inequality names the best filter. Put $\phi_1 = H(f)$ and $\phi_2 = G(f)e^{j2\pi fT}$ into $\phi_1 = k\phi_2^*$.

IN FREQUENCY

$$H_{\text{opt}}(f) = k\, G^*(f)\, e^{-j2\pi fT}$$

The conjugate of $e^{j2\pi fT}$ is $e^{-j2\pi fT}$, and k is the constant of the equality condition.

Take the inverse transform to find the filter in time. The integral that appears is the conjugate of the inverse transform of $G(f)$, read at $T - t$.

IN TIME

$$\begin{aligned} h_{\text{opt}}(t) &= \int_{-\infty}^{\infty} H_{\text{opt}}(f)\, e^{j2\pi ft}\, df \\ &= k \int_{-\infty}^{\infty} G^*(f)\, e^{-j2\pi f(T-t)}\, df \\ &= k\, g^*(T-t) \\ &= k\, g(T-t) \end{aligned}$$

A real pulse has $g^* = g$. The filter is the pulse reversed in time and shifted right by T. It is called the **matched filter**.

For example, the ramp $g(t) = t/T$ on $0 \le t \le T$ with $k = 1$ has $h_{\text{opt}}(0) = g(T-0) = g(T) = 1$.

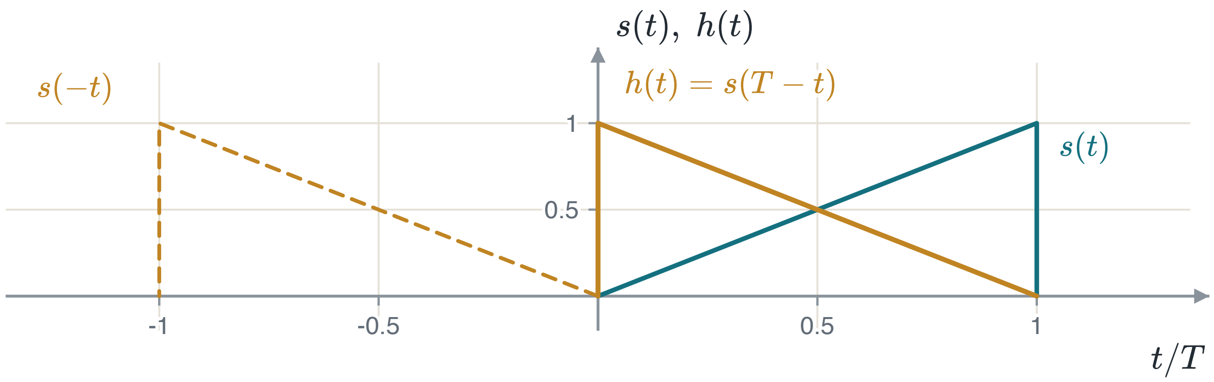


FIGURE 2.4 The matched filter of the ramp $s(t) = t/T$, built in two moves. The ramp (cyan) is reversed in time to $s(-t)$ (amber, dashed), then shifted right by T to $h(t) = s(T-t)$ (amber).

The output of the matched filter

The matched-filter output is a sliding overlap. Write it as a convolution and substitute $h(t) = s(T-t)$, so that $h(t-\tau) = s(T-t+\tau)$.

CONVOLUTION

$$\begin{aligned} y(t) &= \int_{-\infty}^{\infty} s(\tau)\, h(t-\tau)\, d\tau \\ &= \int_{-\infty}^{\infty} s(\tau)\, s(T-t+\tau)\, d\tau \end{aligned}$$

At $t = T$ the two copies of s lie on top of each other. The overlap is then the integral of s^2, which is the energy.

THE PEAK

$$y(T) = \int_{-\infty}^{\infty} s^2(\tau)\, d\tau = E$$

For $s(t) = A$ on $[0, T]$, the overlap at $0 \le t \le T$ has length t. So $y(t) = A^2 t$, and $y(T/2) = A^2 T/2 = E/2$.

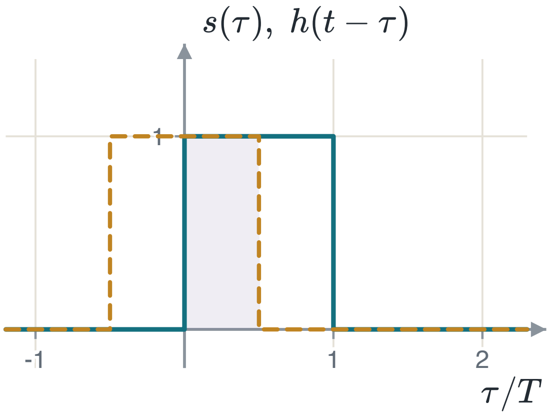


FIGURE 2.5 The overlap at $t = T/2$. The pulse $s(\tau)$ (cyan) and the copy $h(t-\tau)$ (amber, dashed) share the shaded interval.

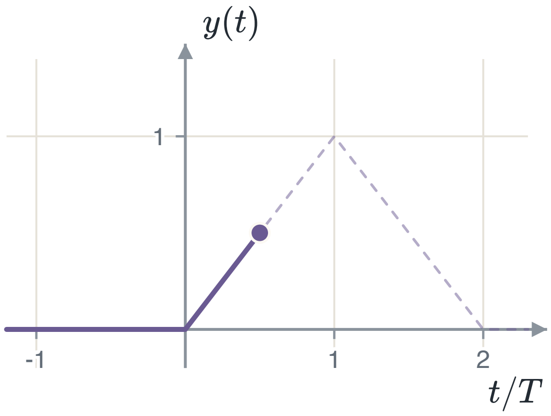


FIGURE 2.6 The output $y(t)$, the area of the overlap, drawn up to $t = T/2$. The dashed rest of the output peaks at $y(T) = E$.

Properties of the matched filter

Two properties follow from the results above.

1. The output is the autocorrelation of the pulse, shifted by T. Its peak is $y(T) = E$.
2. The peak SNR is $2E/N_0$. It depends on the energy, not on the shape.

The second property can be checked in time. Parseval's theorem turns $\int |H(f)|^2 df$ into $\int h^2(t)\,dt$, and h has the same energy as s.

NOISE AT THE OUTPUT

$$
\begin{aligned}
E[n^2(T)] &= \frac{N_0}{2}\int_{-\infty}^{\infty} h^2(t)\,dt \\
&= \frac{N_0}{2}\int_{-\infty}^{\infty} s^2(T-t)\,dt = \frac{N_0 E}{2}
\end{aligned}
$$

Then $\eta = E^2/(N_0E/2) = 2E/N_0$, which is the bound. For example, a rectangular pulse and a half-sine pulse with $E = 10^{-6}$ J and $N_0 = 10^{-7}$ W/Hz both give $\eta_{\max} = 20$.

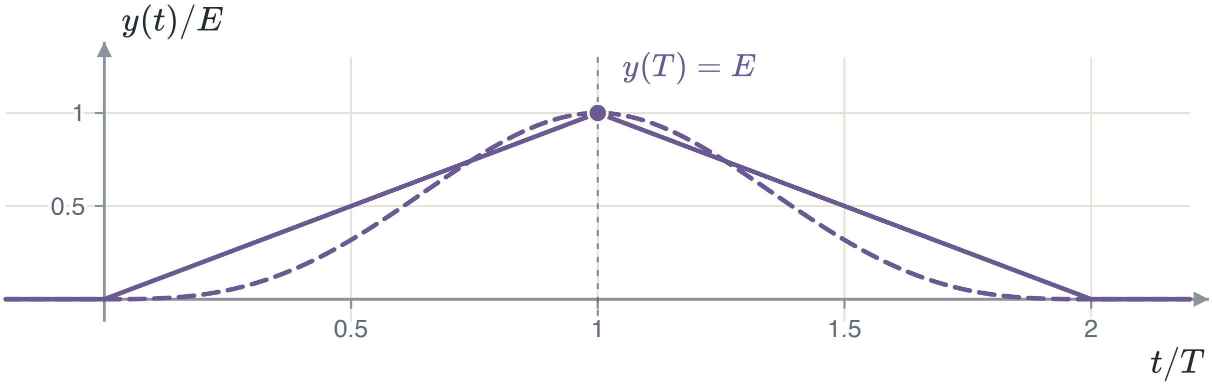


FIGURE 2.7 A rectangular pulse and a half-sine pulse of the same energy E, each through its own matched filter. The rectangular output is solid and the half-sine output dashed. Both peak at $t = T$ with the value E.

EXAMPLE 2.1 – THE MATCHED FILTER OF A RECTANGULAR PULSE

GIVEN $s(t) = A$ on $[0, T]$, with $A = 2$ V, $T = 0.5$ ms and $N_0/2 = 10^{-4}$ W/Hz.

FIND The matched filter $h(t)$, its output $y(t)$ and the largest peak SNR η_{max}.

METHOD Reverse and shift the pulse to get $h(t)$. The output is the sliding overlap of s and h. The largest SNR is $2E/N_0$, which needs only the energy.

SOLUTION $h(t) = s(T - t) = A$ for $0 \le t \le T$. The overlap gives $y(t) = A^2 t$ for $0 \le t \le T$ and $y(t) = A^2(2T - t)$ for $T \le t \le 2T$. The energy is $E = A^2 T = (2)^2(0.5 \times 10^{-3}) = 2 \times 10^{-3}$ J. Then $\eta_{\text{max}} = 2E/N_0 = 2(2 \times 10^{-3})/(2 \times 10^{-4}) = 20$.

CHECK $10 \log_{10} 20 = 13.0$ dB. The peak is $y(T) = A^2 T = E$, as the general result requires.

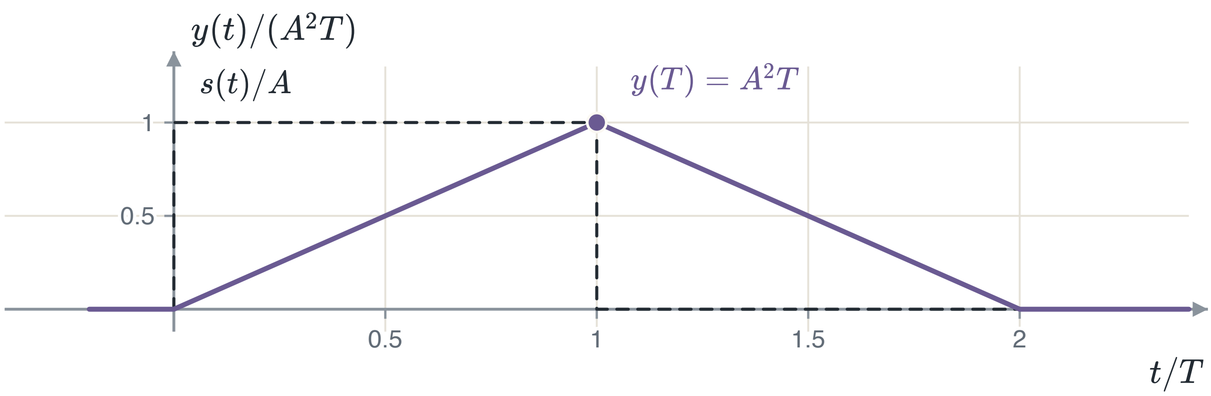


FIGURE 2.8 Example 2.1: the pulse $s(t)/A$ (dashed) and the output of its matched filter, $y(t)/(A^2T)$. The output is a triangle that peaks at $y(T) = A^2T$.

COMMON ERROR

Use $N_0 = 2 \times 10^{-4}$, not $N_0/2$, in $2E/N_0$. The wrong value gives 40.

Matched filters around us

The matched filter appears in radar, Wi-Fi, range finders and serial ports.

- **Radar.** A radar chirp lasts $T = 10$ μs and sweeps $B = 5$ MHz. Its matched output is $(1 - |t|/T)\,|\operatorname{sinc}(Bt(1 - |t|/T))|$, a peak about $1/B = 0.2$ μs wide.
- **Wi-Fi.** Wi-Fi at 1 and 2 Mb/s spreads each bit over the 11-chip Barker code $c[n]$. Its matched output $R[k] = \sum_n c[n]\,c[n + k]$ is 11 at $k = 0$ and 0 or -1 elsewhere.
- **Range finder.** An ultrasonic range finder correlates the echo with the pulse it sent. The peak at $t_0 = 5.83$ ms gives $d = ct_0/2 = 1.00$ m for $c = 343$ m/s.
- **Serial port.** A serial port at 9600 baud integrates each bit over $T_b = 1/9600 = 104.2$ μs, then dumps. The integrator is the matched filter of a rectangular bit.

CORRELATION WITH A TEMPLATE

Each receiver correlates what it hears with a copy of what was sent. That is the matched filter $h(t) = s(T - t)$ read at its peak.

PEAK TIME

The peak falls where the copy lines up with the echo. Its time is the delay, and the delay gives the range.

PULSE COMPRESSION

A long chirp carries much energy, and its matched output is still a peak about $1/B$ wide. The resolution comes from the bandwidth B, not from the length T.

2.2 The demodulator

The unit-energy basis and antipodal signalling

Polar signalling sends $+A$ for a 1 and $-A$ for a 0, over a bit of length T_b. Both waveforms are multiples of one shape, the **basis function** $\psi(t)$.

THE BASIS

$$\psi(t) = \frac{1}{\sqrt{T_b}}, \qquad 0 \le t \le T_b, \qquad \int_0^{T_b} \psi^2(t)\,dt = 1$$

A basis function has unit energy. Outside $[0, T_b]$ it is zero.

Each waveform is then a number times $\psi(t)$. The number is set by the energy per bit E_b.

TWO WAVEFORMS

$$\begin{aligned} s_1(t) &= +A = +\sqrt{E_b}\,\psi(t) \\ s_0(t) &= -A = -\sqrt{E_b}\,\psi(t) \\ E_b &= \int_0^{T_b} A^2\,dt = A^2 T_b \end{aligned}$$

Since $\sqrt{E_b} = A\sqrt{T_b}$, the product $\sqrt{E_b}\,\psi(t)$ is $A\sqrt{T_b}/\sqrt{T_b} = A$. The two waveforms are negatives of each other, so the signalling is called **antipodal**. With $A = 2$ V and $T_b = 1$ ms in a $1\ \Omega$ load, $E_b = (2)^2(10^{-3}) = 4$ mJ.

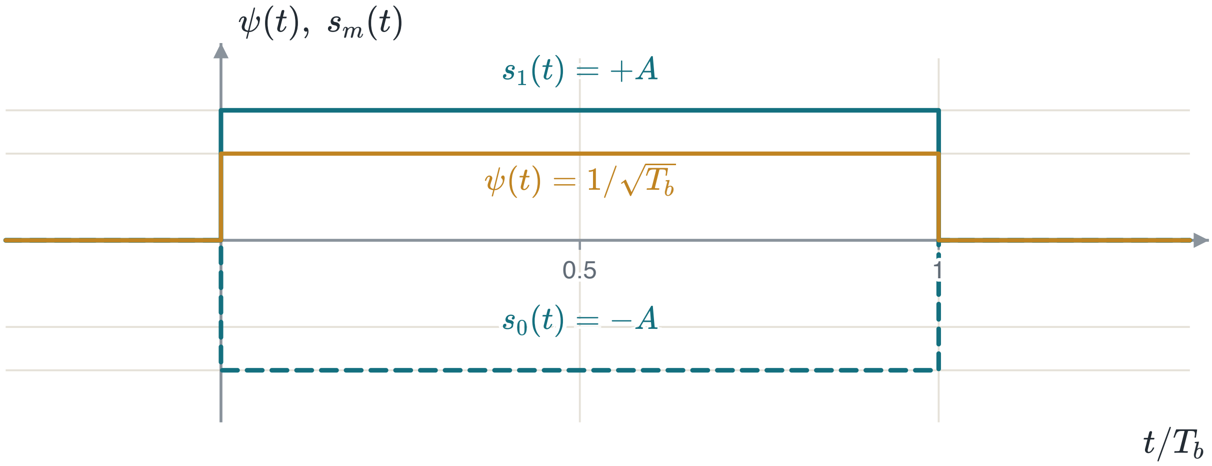


FIGURE 2.9 The basis $\psi(t)$ (amber) and the two waveforms of polar signalling, $s_1(t)$ (cyan) and $s_0(t)$ (cyan, dashed), drawn with $T_b = 1$ and $A = 1.5$. Each waveform is $\psi(t)$ times a number.

Two waveforms as two points

The number that multiplies $\psi(t)$ is found by projection. Multiply the waveform by $\psi(t)$ and integrate over the bit.

COEFFICIENT ON ψ

$$s_m = \int_0^{T_b} s_m(t)\,\psi(t)\,dt$$
$$s_1 = \int_0^{T_b} A\,\frac{1}{\sqrt{T_b}}\,dt = A\sqrt{T_b} = +\sqrt{E_b}$$

For $s_0(t) = -A$ the same steps give $s_0 = -\sqrt{E_b}$. The two waveforms have become two points on one axis, the axis of $\psi(t)$.

The distance between the two points measures how far noise must push the received value to cause an error.

DISTANCE

$$d = s_1 - s_0 = \sqrt{E_b} - \left(-\sqrt{E_b}\right) = 2\sqrt{E_b}$$

A larger distance means noise must move the received value further before it crosses to the wrong point. For $E_b = 4$ mJ, $d = 2\sqrt{4 \times 10^{-3}} = 2(0.0632) = 0.126\ \sqrt{\text{J}}$.

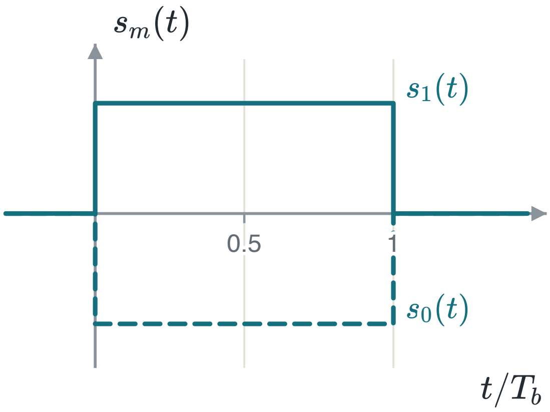


FIGURE 2.10 The two waveforms of polar signalling, $s_1(t)$ (solid) and $s_0(t)$ (dashed).

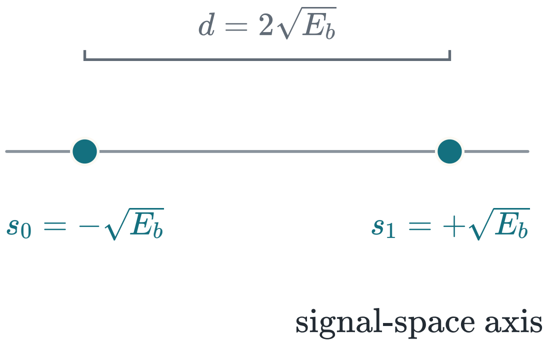


FIGURE 2.11 The same two waveforms as points on the axis of $\psi(t)$. The points are $2\sqrt{E_b}$ apart.

The demodulator output

The demodulator correlates the received signal $x(t) = s_m(t) + w(t)$ with $\psi(t)$. The integral splits into a signal term and a noise term.

THE SAMPLE

$$y(T_b) = \int_0^{T_b} s_m(t)\,\psi(t)\,dt + \int_0^{T_b} w(t)\,\psi(t)\,dt = s_m + n$$

The first integral is the coefficient s_m found above. The second is a noise value n.

The noise value n is Gaussian, because it comes from a linear operation on Gaussian noise. Its mean is zero. Its variance follows from the autocorrelation of white noise.

NOISE VARIANCE

$$\begin{aligned} E[n^2] &= \int_0^{T_b}\int_0^{T_b} E[w(t)w(u)]\,\psi(t)\,\psi(u)\,dt\,du \\ &= \int_0^{T_b}\int_0^{T_b} \frac{N_0}{2}\,\delta(t-u)\,\psi(t)\,\psi(u)\,dt\,du \\ &= \frac{N_0}{2}\int_0^{T_b} \psi^2(t)\,dt = \frac{N_0}{2} \end{aligned}$$

The first line writes n^2 as a double integral and takes the expectation inside. The second uses $R_W(\tau) = (N_0/2)\delta(\tau)$. The impulse then sifts $\psi(u)$ to $\psi(t)$, and $\int \psi^2 dt = 1$.

For example, $N_0 = 2 \times 10^{-4}$ W/Hz gives $\sigma^2 = N_0/2 = 10^{-4}$, so the standard deviation of n is $\sigma = \sqrt{10^{-4}} = 0.01$.

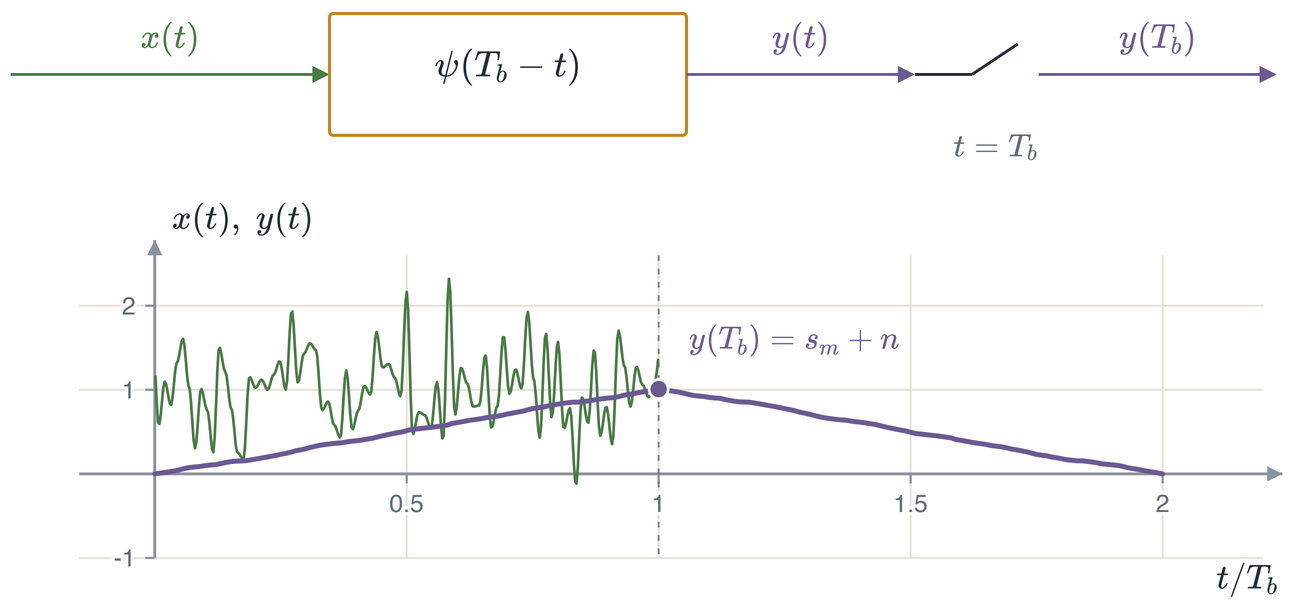


FIGURE 2.12 The filter matched to $\psi(t)$, and what it does to one noisy bit. The received bit is green and the filter output violet. At $t = T_b$ the output is the symbol s_m plus a noise value n.

Correlator and matched-filter demodulators

The number y can be formed in two ways. The **correlator** multiplies by $\psi(t)$ and integrates over the bit. It reads the result at T_b, then resets.

CORRELATOR

$$y = \int_0^{T_b} x(t)\,\psi(t)\,dt$$

The **matched filter** has the impulse response $\psi(T_b - t)$. Write its output as a convolution and put $t = T_b$
.

MATCHED FILTER AT T_b

$$\begin{aligned} h(t) &= \psi(T_b - t) \\ y(t) &= \int_{-\infty}^{\infty} x(\tau)\,\psi(T_b - t + \tau)\,d\tau \\ y(T_b) &= \int_0^{T_b} x(\tau)\,\psi(\tau)\,d\tau \end{aligned}$$

The last line is the correlator output, for any shape of ψ. The two demodulators agree at $t = T_b$.

Away from T_b the two outputs differ. Take one noiseless bit with $s_m = 1$ and the rectangular ψ. On $[0, T_b]$ both outputs rise along the same line.

After T_b the correlator has been reset, while the matched-filter output falls as $2 - t/T_b$. So $y(1.5T_b) = 0.5$ at the filter. Only the value at T_b is used.

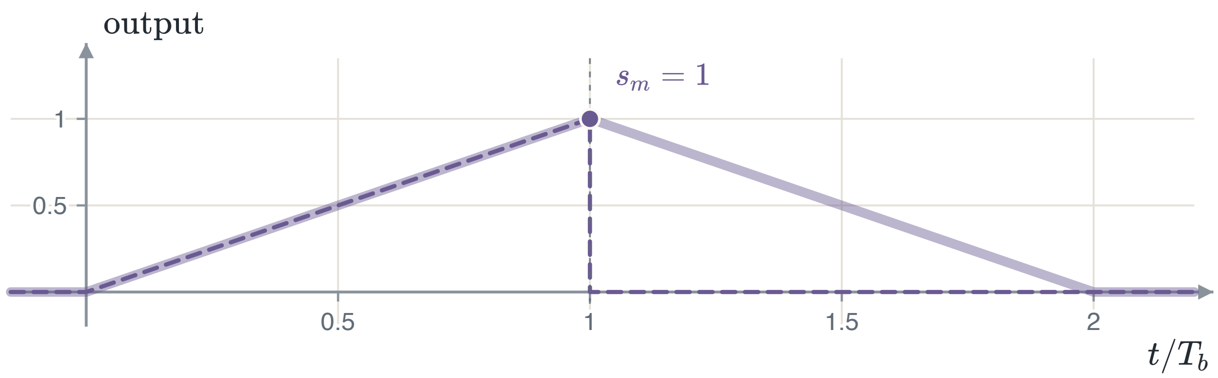


FIGURE 2.13 One noiseless bit with $s_m = 1$ and the rectangular ψ. The matched-filter output is the wide line and the correlator output is dashed. The correlator integrates from 0 and is reset at T_b. At $t = T_b$ both read s_m.

Demodulators around us

Antipodal signalling and the correlator appear in serial links and digital receivers.

- **RS-485.** An RS-485 pair at 1 Mb/s sends $s(t) = \pm A$ on the difference of two wires, with $A = 2$ V. That is polar signalling with $s_m = \pm A\sqrt{T_b}$.
- **UART.** A UART takes 16 samples a bit. The sum $y = \sum_{n=0}^{15} x[n]$ is a correlator with a rectangular ψ, run on samples.
- **Decision statistic.** The values of y for 4000 bits at $E_b/N_0 = 6$ dB form two bumps. They sit at $\pm\sqrt{E_b}$, each with spread $\sigma = \sqrt{N_0/2}$.
- **Sampling instant.** Sampled $0.1T_b$ early, the output keeps $0.9\sqrt{E_b}$. The signal energy seen drops to $0.81E_b$, a loss of 0.92 dB.

> ⓘ TWO POINTS ON A LINE
>
> Every antipodal link sends $\pm A$ for one bit. After the demodulator only the number $y = \pm\sqrt{E_b} + n$ is left.

> ⓘ A CORRELATOR ON SAMPLES
>
> A digital receiver adds samples instead of integrating. The sum of N samples a bit is the correlator with ψ sampled N times.

> ⚠ TIMING
>
> The correlator is read at the end of the bit. An offset of ϵ scales the signal term by $1 - \epsilon/T_b$.

2.3 The decision and its error

The decision statistic

The demodulator output y at $t = T_b$ is the **decision statistic**. It is all the receiver keeps of the bit.

THE STATISTIC

$$y = s_m + n, \qquad s_m = \pm\sqrt{E_b}, \qquad n \sim \mathcal{N}\left(0, \tfrac{N_0}{2}\right)$$

The symbol sets the mean of y. The noise sets its spread.

Given the symbol, y is Gaussian with mean s_m and variance $\sigma^2 = N_0/2$. Put these into the Gaussian density.

CONDITIONAL DENSITY

$$\begin{aligned} f_Y(y \mid s_1) &= \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y-\sqrt{E_b})^2}{2\sigma^2}\right) \\ &= \frac{1}{\sqrt{\pi N_0}} \exp\left(-\frac{(y-\sqrt{E_b})^2}{N_0}\right) \end{aligned}$$

The second line uses $2\sigma^2 = N_0$. For s_0, replace $-\sqrt{E_b}$ by $+\sqrt{E_b}$ in the exponent. With $E_b = 1$ and $N_0 = 0.5$, $\sigma = \sqrt{0.25} = 0.5$ for either symbol.

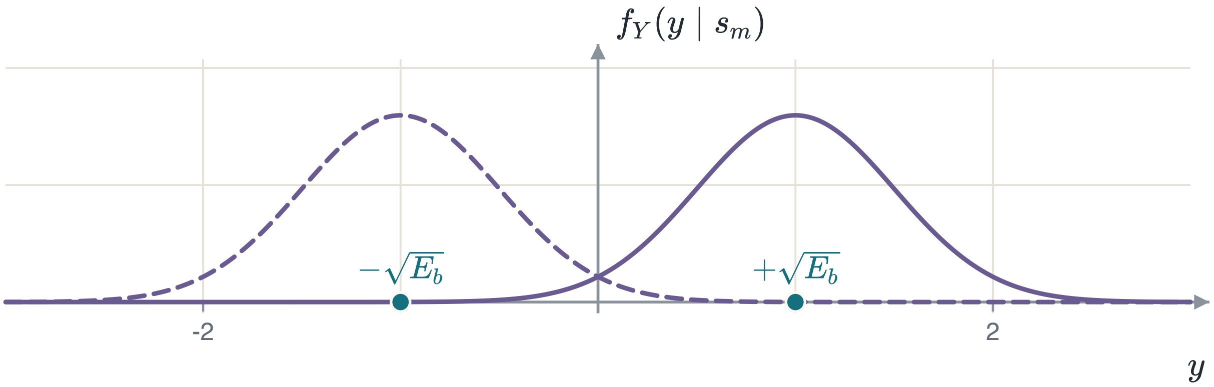


FIGURE 2.14 The density of y for each symbol, drawn with $E_b = 1$ and $N_0 = 0.5$. The density given s_0 is dashed and the density given s_1 solid. Each is a Gaussian centred on its symbol, and the two overlap.

The two conditional errors

The receiver decides s_1 when $y > \lambda$ and s_0 otherwise. Each symbol then has its own error probability: the area of its density on the wrong side of λ.

ERROR GIVEN s_0

$$\begin{aligned} P(\mathrm{err} \mid s_0) &= \int_{\lambda}^{\infty} \frac{1}{\sqrt{\pi N_0}}\, e^{-(y+\sqrt{E_b})^2/N_0}\, dy \\ &= \int_{(\lambda+\sqrt{E_b})/\sigma}^{\infty} \frac{1}{\sqrt{2\pi}}\, e^{-z^2/2}\, dz \\ &= Q\left(\frac{\lambda+\sqrt{E_b}}{\sigma}\right) \end{aligned}$$

Substitute $z = (y+\sqrt{E_b})/\sigma$ with $\sigma = \sqrt{N_0/2}$. The lower limit $y = \lambda$ becomes $z = (\lambda+\sqrt{E_b})/\sigma$. Since $dy = \sigma\, dz$ and $\sigma/\sqrt{\pi N_0} = 1/\sqrt{2\pi}$, the integrand becomes the unit Gaussian density.

ERROR GIVEN s_1

$$P(\text{err} \mid s_1) = \int_{-\infty}^{\lambda} f_Y(y \mid s_1)\, dy = Q\left(\frac{\sqrt{E_b} - \lambda}{\sigma}\right)$$

The same substitution with $z = (\sqrt{E_b} - y)/\sigma$. Here $Q(x) = \frac{1}{2}\operatorname{erfc}(x/\sqrt{2})$ is the Gaussian tail. For example, $E_b = 1$, $\sigma = 0.5$ and $\lambda = 0$ give $P(\text{err} \mid s_0) = Q\big((0+1)/0.5\big) = Q(2) = 0.0228$.

Moving λ trades one error against the other. One area grows as the other shrinks.

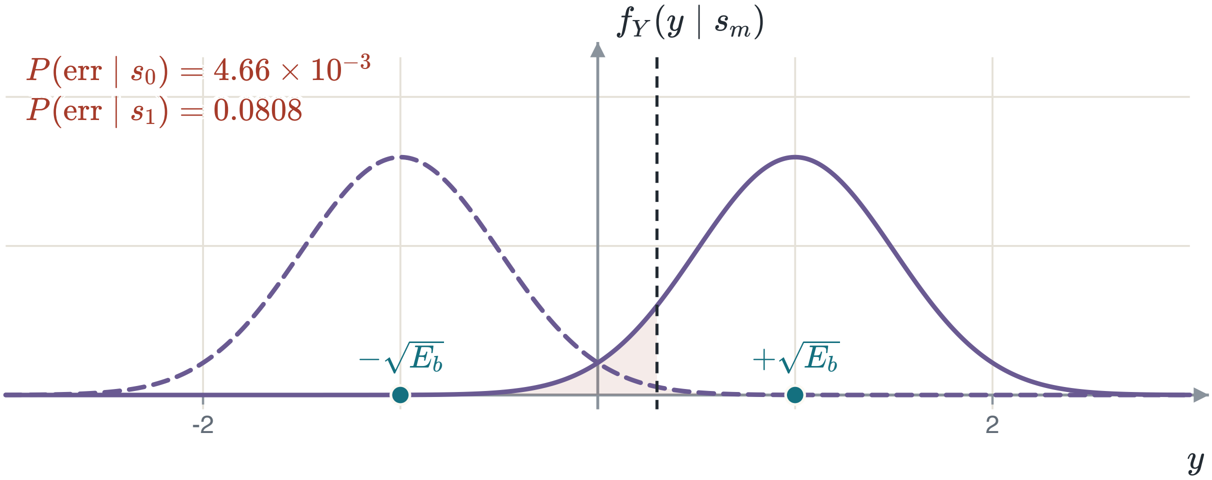


FIGURE 2.15 The two conditional densities with $E_b = 1$ and $N_0 = 0.5$, and the threshold at $\lambda = 0.3$. Each red area is one conditional error. Moving λ to the right shrinks the error given s_0 and grows the error given s_1.

The optimal threshold

The best threshold minimises the average error probability. The priors $p_0 = P(s_0)$ and $p_1 = P(s_1)$ weight the two conditional errors.

MINIMISE THE AVERAGE ERROR

$$\begin{aligned} P_e(\lambda) &= p_0\, P(\text{err} \mid s_0) + p_1\, P(\text{err} \mid s_1) \\ \frac{dP_e}{d\lambda} &= -p_0\, f_Y(\lambda \mid s_0) + p_1\, f_Y(\lambda \mid s_1) = 0 \end{aligned}$$

Differentiate each integral with respect to its limit λ. The error given s_0 has λ as its lower limit, so its derivative carries a minus sign.

At the minimum the two weighted densities are equal. Cancel the common factor $1/\sqrt{\pi N_0}$, take logarithms, then expand the two squares.

$$\begin{aligned} p_0\, e^{-(\lambda+\sqrt{E_b})^2/N_0} &= p_1\, e^{-(\lambda-\sqrt{E_b})^2/N_0} \\ \ln p_0 - \frac{(\lambda+\sqrt{E_b})^2}{N_0} &= \ln p_1 - \frac{(\lambda-\sqrt{E_b})^2}{N_0} \\ (\lambda+\sqrt{E_b})^2 - (\lambda-\sqrt{E_b})^2 &= N_0 \ln\frac{p_0}{p_1} \\ 4\lambda\sqrt{E_b} &= N_0 \ln\frac{p_0}{p_1} \end{aligned}$$

The λ^2 and E_b terms cancel in the difference of the two squares. Divide by $4\sqrt{E_b}$.

THE OPTIMAL THRESHOLD

$$\lambda_{\text{opt}} = \frac{N_0}{4\sqrt{E_b}} \ln \frac{p_0}{p_1}$$

For example, $p_0 = 0.7$, $E_b = 1$ and $N_0 = 0.5$ give $\lambda_{\text{opt}} = \frac{0.5}{4} \ln \frac{0.7}{0.3} = 0.125(0.847) = 0.106$. It moves toward the less likely symbol.

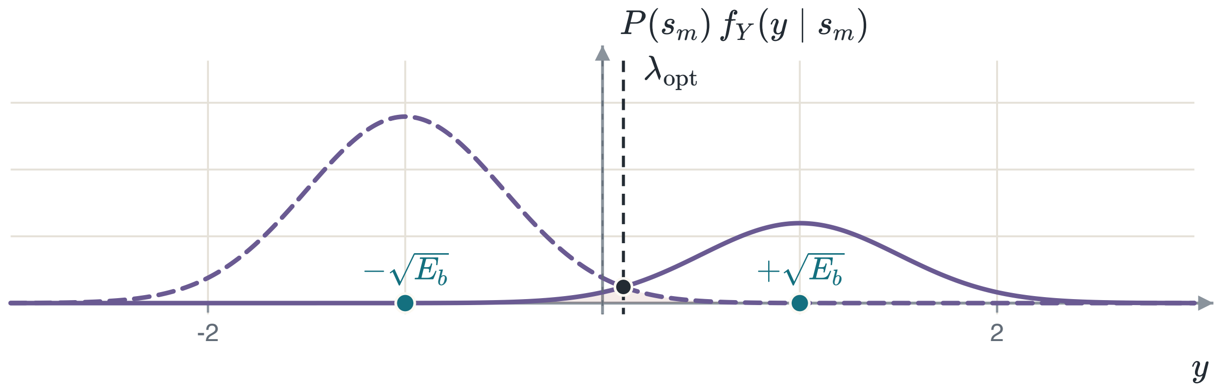


FIGURE 2.16 The two densities weighted by their priors, for $p_0 = 0.7$, $E_b = 1$ and $N_0 = 0.5$. The weighted density of s_0 is dashed. They cross at λ_{opt}, right of zero. The shaded area is P_e.

Threshold, priors and noise

The formula for λ_{opt} shows two effects.

1. The threshold moves away from the more likely symbol, so that symbol gets the larger region.
2. The move grows with N_0. With little noise, y decides and the prior hardly matters.

With equal priors, $p_0 = p_1 = 0.5$, the logarithm is $\ln(0.5/0.5) = \ln 1 = 0$. Then $\lambda_{\text{opt}} = 0$ for every N_0
.

ⓧ COMMON ERROR

Use $\ln(p_0/p_1)$, not $\ln(p_1/p_0)$. With $p_0 > p_1$ the threshold is positive, so the more likely s_0 keeps more of the axis.

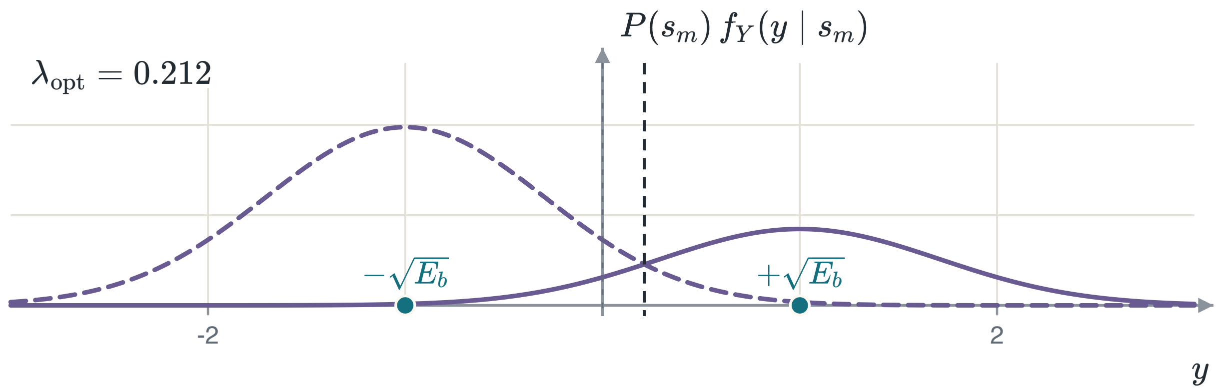


FIGURE 2.17 The weighted densities for $p_0 = 0.7$ at $E_b/N_0 = 0$ dB, with $E_b = 1$. The noise density is twice that of the last figure, and the threshold has moved twice as far, to $\lambda_{\text{opt}} = 0.212$.

The Q function

The Gaussian tail has no closed form. It is written as a function of its own, $Q(x)$.

DEFINITION

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^{\infty} e^{-z^2/2}\, dz = \tfrac{1}{2}\,\mathrm{erfc}\left(\frac{x}{\sqrt{2}}\right)$$

$Q(x)$ is the probability that a Gaussian of mean 0 and variance 1 exceeds x.

A few values recur throughout the course.

VALUES

$$Q(0) = 0.5, \quad Q(1) = 0.159, \quad Q(2) = 0.0228, \quad Q(3) = 1.35 \times 10^{-3}$$

For large x, $Q(x) \le \frac{1}{2} e^{-x^2/2}$, so the tail falls faster than any power of x.

A Gaussian value of mean 0 and standard deviation σ is divided by σ before Q is used. The probability that it exceeds 3σ is $Q(3) = 1.35 \times 10^{-3}$.

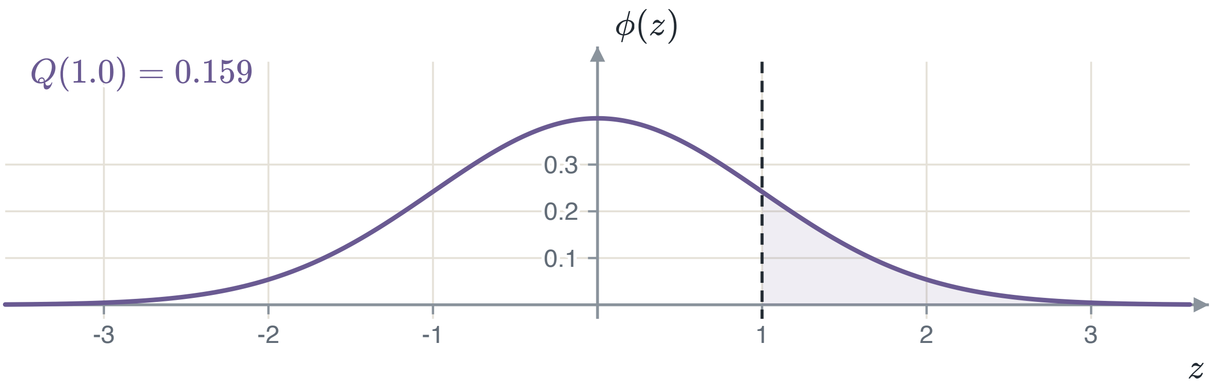


FIGURE 2.18 $Q(x)$ is the shaded area under the standard normal density $\phi(z)$ to the right of x, here at $x = 1$.

The bit error probability

With equal priors the threshold is $\lambda = 0$. Put $\lambda = 0$ in both conditional errors, and the two become equal.

EQUAL PRIORS

$$P_b = \tfrac{1}{2} Q\left(\frac{\sqrt{E_b}}{\sigma}\right) + \tfrac{1}{2} Q\left(\frac{\sqrt{E_b}}{\sigma}\right) = Q\left(\frac{\sqrt{E_b}}{\sigma}\right)$$

$$\frac{\sqrt{E_b}}{\sigma} = \frac{\sqrt{E_b}}{\sqrt{N_0/2}} = \sqrt{\frac{2E_b}{N_0}}$$

The result is the bit error probability of polar signalling.

POLAR SIGNALLING

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

$2E_b/N_0$ is the peak SNR of the matched filter, so $P_b = Q(\sqrt{\eta_{\max}})$. For example, $E_b/N_0 = 4$, which is 6.02 dB, gives $P_b = Q(\sqrt{8}) = Q(2.83) = 2.3 \times 10^{-3}$. Dropping the factor 2 gives $Q(2) = 2.3 \times 10^{-2}$.

The error probability falls slowly at low E_b/N_0 and steeply at high E_b/N_0.

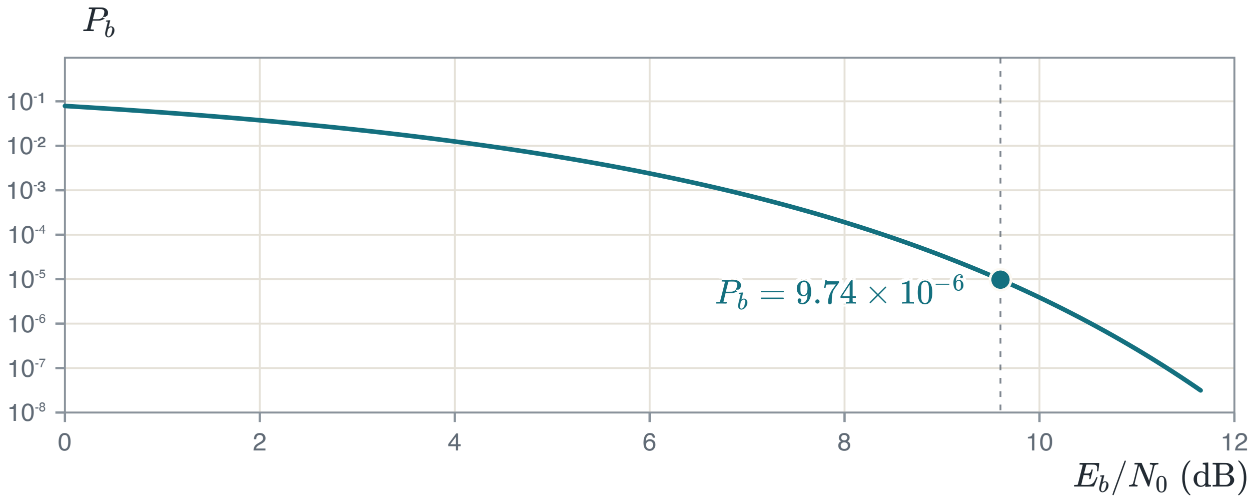


FIGURE 2.19 Bit error probability of polar signalling against E_b/N_0. The marked point is $E_b/N_0 = 9.6$ dB, where $P_b = 9.74 \times 10^{-6}$.

EXAMPLE 2.2 — UNEQUAL PRIORS: THE THRESHOLD

GIVEN Polar signalling with $E_b = 1$, $N_0 = 0.1$ and $P(s_1) = 0.3$.

FIND (a) The optimal threshold λ_{opt}.

METHOD The priors are unequal, so the threshold is not zero. Use $\lambda_{\text{opt}} = \frac{N_0}{4\sqrt{E_b}} \ln \frac{p_0}{p_1}$ with $p_0 = 1 - P(s_1) = 0.7$ and $p_1 = 0.3$.

SOLUTION $\lambda_{\text{opt}} = \frac{0.1}{4(1)} \ln \frac{0.7}{0.3} = 0.025(0.8473) = 0.0212$.

CHECK Both weighted densities are equal there. $p_0 f_Y(\lambda \mid s_0) = \frac{0.7}{\sqrt{0.1\pi}} e^{-(1.0212)^2/0.1} = 1.249\, e^{-10.428} = 3.70 \times 10^{-5}$. $p_1 f_Y(\lambda \mid s_1) = \frac{0.3}{\sqrt{0.1\pi}} e^{-(0.9788)^2/0.1} = 0.535\, e^{-9.581} = 3.70 \times 10^{-5}$.

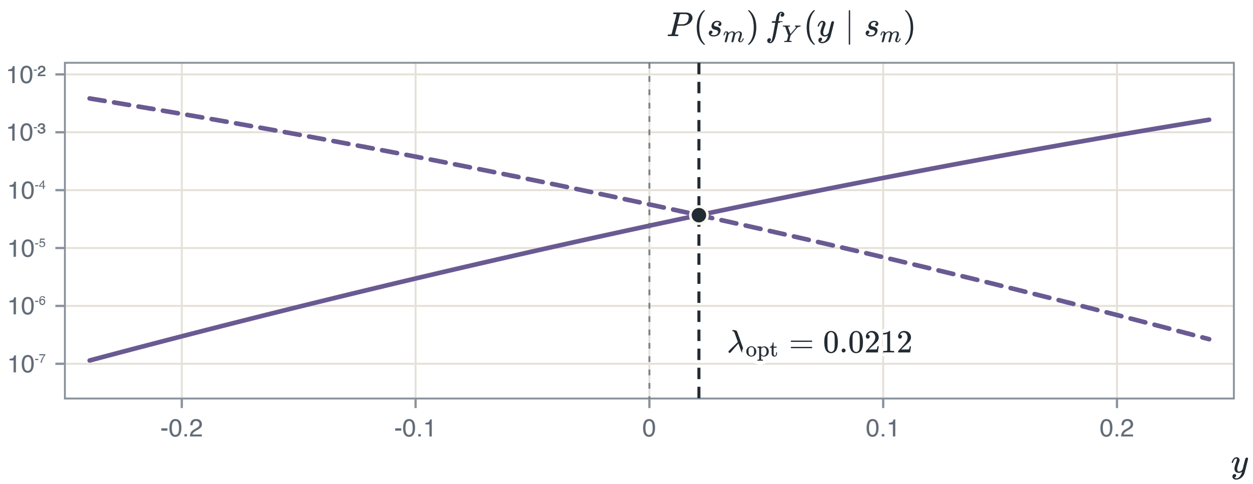


FIGURE 2.20 Example 2.2: the two weighted densities on a logarithmic scale, where each is a parabola. The weighted density of s_0 is dashed. They cross at $\lambda_{\text{opt}} = 0.0212$, just right of zero.

> ⊗ COMMON ERROR
>
> Use $p_0 = 0.7$, not the given 0.3, for s_0. Swapping them gives -0.0212.

EXAMPLE 2.3 — UNEQUAL PRIORS: THE ERROR PROBABILITY

GIVEN $E_b = 1$, $N_0 = 0.1$, $p_0 = 0.7$ and $\lambda_{\text{opt}} = 0.0212$, from Example 2.2.

FIND (b) P_e at λ_{opt}, against P_e at $\lambda = 0$.

METHOD Weight the two conditional errors by their priors: $P_e = p_0\, Q\big((\lambda + \sqrt{E_b})/\sigma\big) + p_1\, Q\big((\sqrt{E_b} - \lambda)/\sigma\big)$, with $\sigma = \sqrt{N_0/2} = 0.2236$.

SOLUTION $P(\text{err} \mid s_0) = Q(1.0212/0.2236) = Q(4.567) = 2.48 \times 10^{-6}$. $P(\text{err} \mid s_1) = Q(0.9788/0.2236) = Q(4.377) = 6.01 \times 10^{-6}$. In units of 10^{-6}, $P_e = 0.7(2.48) + 0.3(6.01) = 3.53$, so $P_e = 3.53 \times 10^{-6}$.

CHECK At $\lambda = 0$ both conditional errors are $Q(1/0.2236) = Q(4.472)$, so $P_e = Q(4.472) = 3.87 \times 10^{-6}$. The ratio $3.87/3.53 = 1.10$, so the zero threshold is about 10 per cent worse.

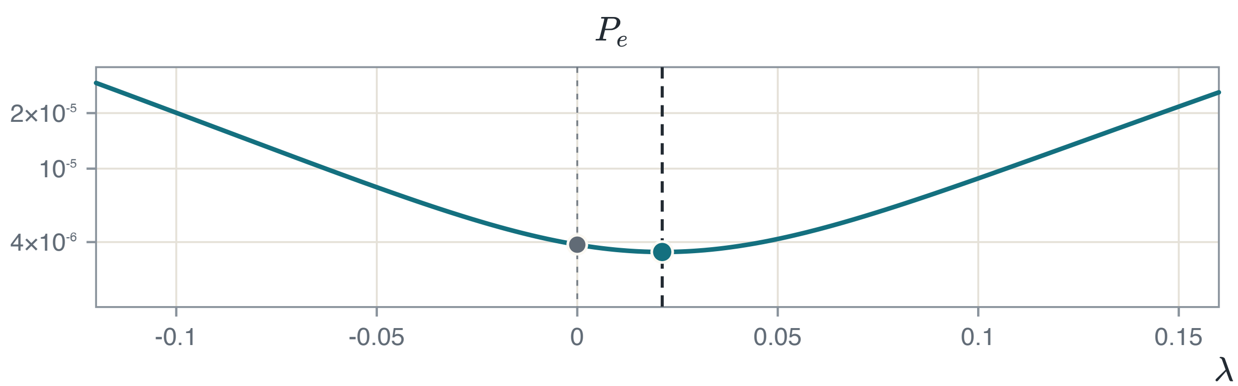


FIGURE 2.21 Example 2.3: P_e against the threshold for the same link. The minimum sits at $\lambda_{\text{opt}} = 0.0212$. At $\lambda = 0$ (grey dot) it is about 10 per cent higher.

COMMON ERROR

Use $\sigma = \sqrt{N_0/2}$, not $\sqrt{N_0}$. The wrong σ raises P_e over a hundred times.

Error rates around us

The error probability sets link targets, moves the threshold for rare events, and is measured by counting errors.

- **Ethernet.** Wired Ethernet asks for $P_b < 10^{-10}$. Polar signalling reaches it at $E_b/N_0 = 13.1$ dB, where $\sqrt{2E_b/N_0} = 6.36$.
- **Rare events.** A sensor reports a rare event with prior $p_1 = 0.1$. With $E_b = 1$ and $N_0 = 0.5$, $\lambda_{\text{opt}} = 0.125 \ln 9 = 0.275$, moved toward the rare symbol.
- **The tail on decades.** On a logarithmic scale, the Gaussian tail $Q(x) = \frac{1}{2}\,\mathrm{erfc}(x/\sqrt{2})$ keeps steepening. Above $x = 3$ each unit step in x divides $Q(x)$ by more than 40.
- **Error counts.** A tester counts errors in blocks of 10^5 bits at $P_b = 10^{-4}$. It expects 10 a block, and the counts scatter around that.

TARGET ERROR RATE

A standard fixes the largest P_b a link may have. The curve $Q\left(\sqrt{2E_b/N_0}\right)$ turns that target into the E_b/N_0 the link needs.

COUNTING ERRORS

A measured P_b is errors divided by bits. About 10 errors are needed for a steady estimate, so $P_b = 10^{-4}$ needs about 10^5 bits.

RARE SYMBOLS

When one symbol is rare, λ_{opt} moves toward it. The receiver then needs stronger evidence before it decides the rare symbol.

2.4 Intersymbol interference

Intersymbol interference

A first-order RC lowpass is the simplest bandlimited channel. It has one parameter, the 3 dB bandwidth B.

THE RC CHANNEL

$$H(f) = \frac{1}{1 + jf/B}, \qquad h(t) = \frac{1}{\tau} e^{-t/\tau}\ (t \geq 0), \qquad \tau = \frac{1}{2\pi B}$$

The channel responds to a sudden change with the time constant τ.

Send one unit bit on $[0, T_b]$. Inside the bit the output is the integral of h. It never reaches 1 inside the bit, and it decays after the bit ends.

ONE BIT

$$r(t) = \int_0^t \frac{1}{\tau} e^{-u/\tau}\, du = 1 - e^{-t/\tau}, \qquad 0 \le t \le T_b$$
$$r(T_b) = 1 - q, \qquad q = e^{-T_b/\tau} = e^{-2\pi B T_b}$$

The share q is the part of the bit's level still missing at the end of the bit. For $BT_b = 0.25$, $q = e^{-2\pi(0.25)} = e^{-\pi/2} = 0.208$ and $r(T_b) = 1 - q = 0.792$.

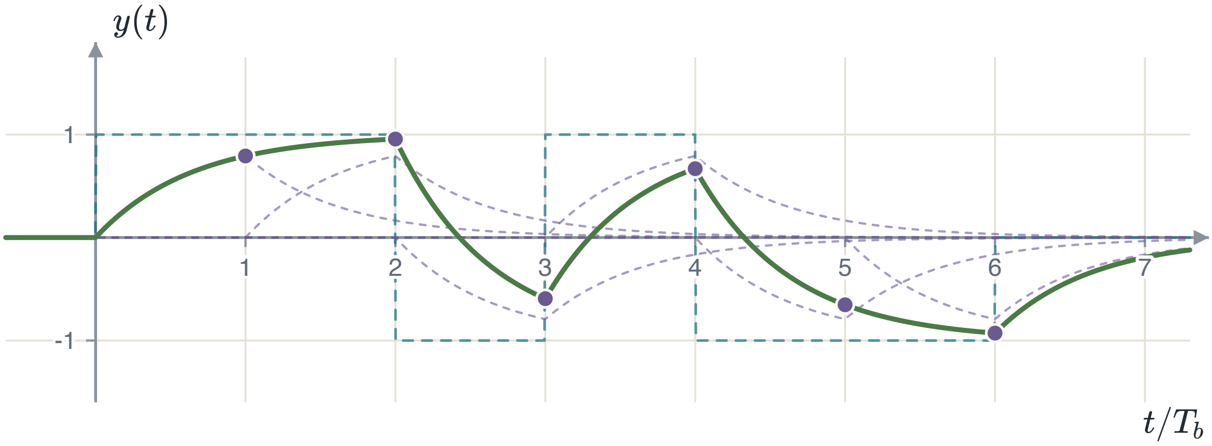


FIGURE 2.22 Six bits $1, 1, 0, 1, 0, 0$ through an RC channel with $BT_b = 0.25$. The sent bits are dashed cyan and each bit's own response is dashed violet. Their sum is the received waveform (green), and the dots are its samples at the end of each bit.

The interference term

After its bit ends, a response decays as $(1-q)e^{-(t-T_b)/\tau}$. One bit later it is $(1-q)q$, and m bits later $(1-q)q^m$.

ONE SAMPLE

$$y_k = a_k(1-q) + \sum_{m=1}^{\infty} a_{k-m}\,(1-q)\,q^m, \qquad a_k = \pm 1$$

Each earlier bit a_{k-m} adds its tail to the sample of bit k. The sum of the tails is the **intersymbol interference**.

The interference is largest when every earlier bit has the sign opposite to the current one. Its size is then a geometric series.

WORST CASE

$$\sum_{m=1}^{\infty}(1-q)\,q^m = (1-q)\,\frac{q}{1-q} = q$$

The sample then shrinks from $1-q$ to $(1-q)-q = 1-2q$. For example, a 1 after a long run of 0s at $BT_b = 0.25$ gives $1 - 2(0.208) = 0.584$.

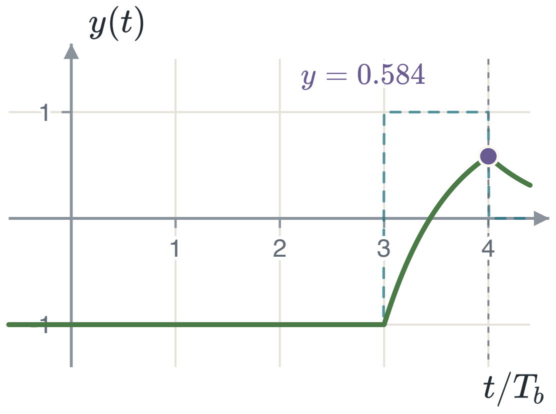


FIGURE 2.23 The bits $0, 0, 0, 1$ after a line idle at the level of a 0, with $BT_b = 0.25$. The last sample is marked.

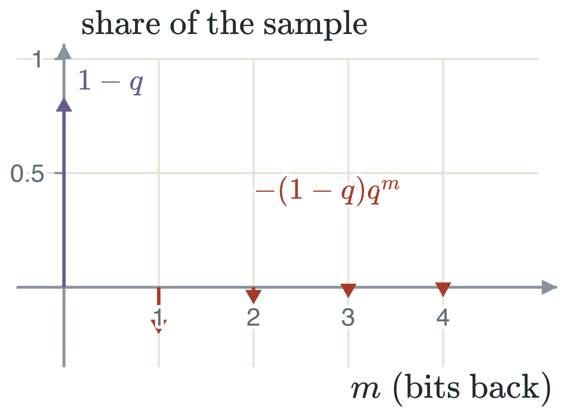


FIGURE 2.24 The last sample split into its own share $1-q$ (violet) and the tails $-(1-q)q^m$ of the earlier bits (red).

EXAMPLE 2.4 — THE SAMPLES OF A BIT PATTERN

GIVEN Bits $0\,0\,0\,1\,1\,1\,0\,1$ sent after a line idle at the level of a 0 ($a_k = -1$), through an RC channel with $BT_b = 0.25$, so $q = 0.208$.

FIND The sample at the end of the seventh bit, y_7.

METHOD Every earlier bit leaves a tail in the sample. Add the bit's own share, the tails of the three 1s, and the tails of every earlier bit and the idle line, all -1.

SOLUTION $y_7 = -(1-q) + (1-q)(q + q^2 + q^3) - (1-q)\sum_{m=4}^{\infty} q^m$. The middle term is $q - q^4$ and the last is q^4, by the geometric series. So $y_7 = -(1-q) + (q - q^4) - q^4 = -1 + 2q - 2q^4 = -1 + 2(0.2079) - 2(0.0019) = -0.588$.

CHECK The sample is negative, so the 0 is still decided correctly.

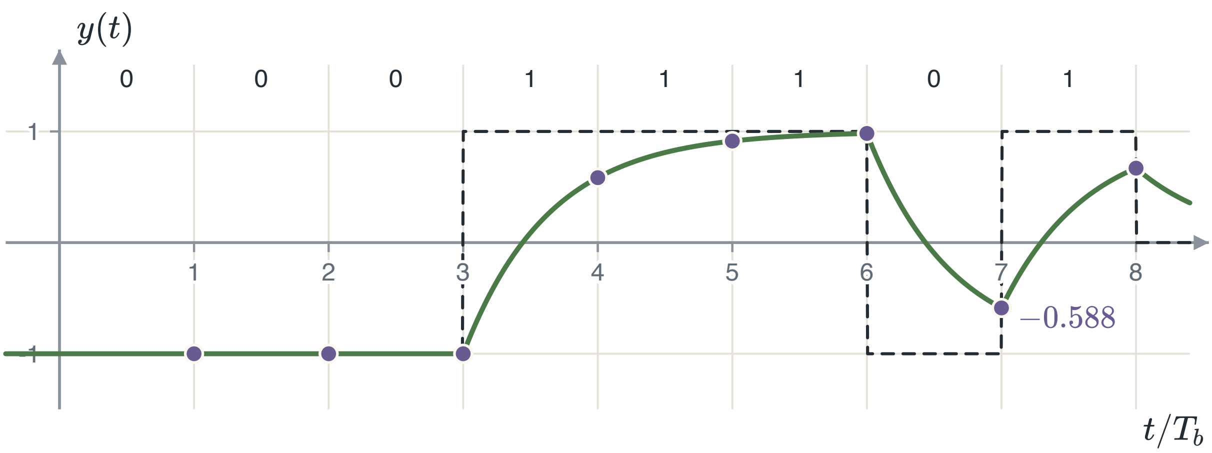


FIGURE 2.25 Example 2.4: the sent pattern (dashed) and the channel output (green), with its value at the end of each bit. The seventh sample is -0.588.

COMMON ERROR

Use the full history, not only the previous bit. Each earlier bit still contributes its own tail $(1-q)q^m$.

The eye diagram

An **eye diagram** shows the interference of every bit pattern at once. Cut the received waveform into pieces $2T_b$ long, centred on the sampling instants, and draw them all on one axis.

The vertical gap at the sampling instant is the **opening** of the eye. The worst pattern has every earlier bit opposite to the current one.

OPENING

$$\min_{\text{patterns}} |y_k| = (1-q) - q = 1 - 2q, \qquad \text{opening} = 2(1-2q)$$

For $BT_b = 0.3$, $q = e^{-0.6\pi} = 0.152$, so the opening is $2(1-2q) = 2(0.696) = 1.39$.

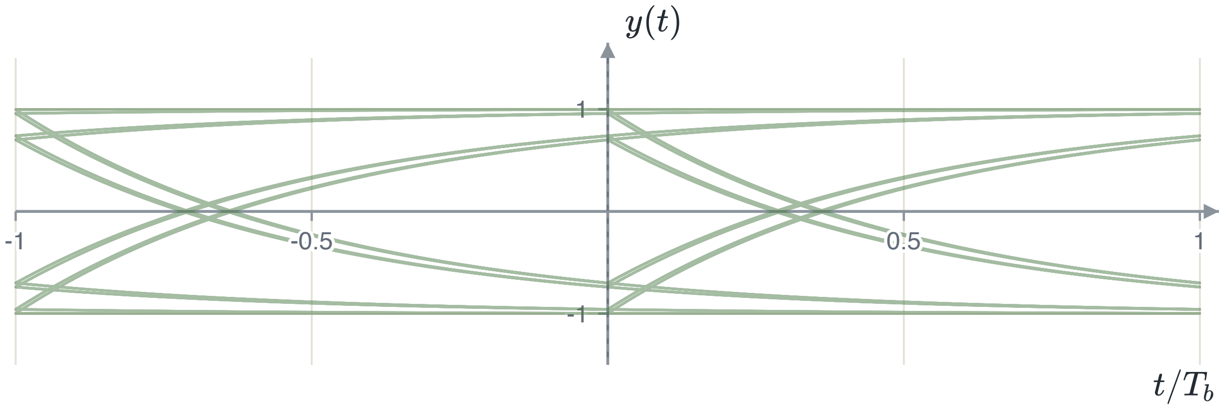


FIGURE 2.26 The eye of the RC channel with $BT_b = 0.3$. Each trace is the received waveform over two bits, centred on a sampling instant, for one of 128 bit patterns. Laid on top of each other they form the eye.

Eye closure

A narrower channel closes the eye. It is fully closed when the opening reaches zero. Set $1 - 2q = 0$ and take logarithms.

CLOSURE

$$\begin{aligned} 1 - 2q &= 0 \\ e^{-2\pi BT_b} &= \tfrac{1}{2} \\ BT_b &= \frac{\ln 2}{2\pi} = 0.110 \end{aligned}$$

Below $BT_b = 0.110$ some patterns cross zero at the sampling instant, and decisions fail with no noise at all. At $BT_b = 0.1$, $q = e^{-0.2\pi} = 0.533 > 0.5$, so $1 - 2q = -0.067 < 0$ and the eye is closed.

NOISE MARGIN

Noise must move a sample by more than $1 - 2q$ to cause an error. A half-closed eye halves the margin, which costs 6 dB of SNR.

COMMON ERROR

Use the worst pattern, not a typical one. A random stream may run for a long time before it shows the pattern that closes the eye.

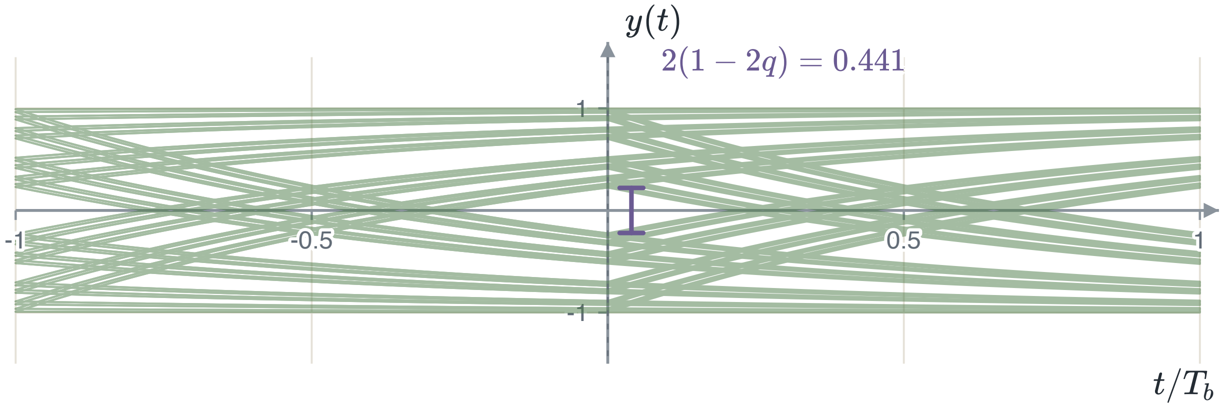


FIGURE 2.27 The eye at $BT_b = 0.15$, where $q = 0.390$ and the opening has shrunk to 0.441. Below $BT_b = 0.110$ the eye is closed.

Noise on the eye

A real sample carries noise as well as interference. Each sample then has three parts.

ONE SAMPLE

$$y_k = \underbrace{a_k(1-q)}_{\text{own bit}} + \underbrace{\mathrm{ISI}_k}_{\text{earlier bits}} + \underbrace{n_k}_{\text{noise}}$$

Interference moves each sample to one of a few levels. Noise then spreads every level into a bell.

WORST PATTERN

The levels nearest zero fail first. Their margin is $1-2q$, not $1-q$, so a half-closed eye needs much less noise to fail.

As the noise grows from zero, the worst-pattern samples cross zero first. At $BT_b = 0.3$ they sit at $\pm(1-2q) = \pm 0.70$, the levels closest to the threshold.

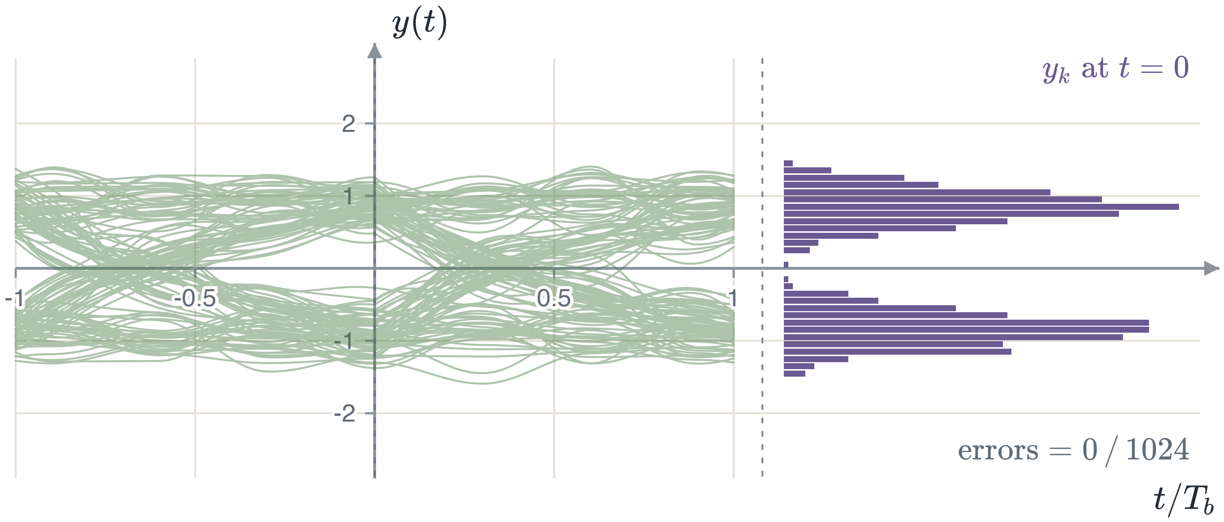


FIGURE 2.28 The eye at $BT_b = 0.3$ with noise of standard deviation $\sigma = 0.2$. The bars at the right count 1024 samples taken at $t = 0$. Violet bars fall on the correct side of zero, red bars on the wrong side.

Intersymbol interference around us

Intersymbol interference appears in cables, radio echoes and optical fibre.

- **Cable.** A long twisted pair acts like the RC channel with $BT_b = 0.15$. One bit leaves $(1-q)q^m$ in the sample m bits later, with $q = e^{-2\pi BT_b} = 0.39$.
- **Radio echo.** An echo three symbols late with gain 0.6 gives $h[n] = \delta[n] + 0.6\,\delta[n-3]$. Each sample then carries $0.6\,a_{k-3}$ from an older symbol.
- **Optical fibre.** A 10 Gb/s light pulse with $\sigma_0 = 20$ ps spreads to $\sigma = \sqrt{20^2 + 34^2} = 39.4$ ps after 20 km of fibre. It then reaches into the next bit, 100 ps away.
- **Eye opening.** The eye opening $2(1-2q)$ of the RC channel shrinks as BT_b falls. It reaches zero at $BT_b = \ln 2/(2\pi) = 0.110$.

> ⓘ CHANNEL MEMORY
>
> Every bandlimited channel spreads a pulse beyond its own bit. The sample of one bit then carries part of its neighbours.

> ⓘ ECHOES
>
> A radio echo is intersymbol interference with a delay. An echo d symbols late with gain g adds $g\,a_{k-d}$ to y_k.

> ⚠ ERRORS WITHOUT NOISE
>
> When the eye is closed, some bit patterns are decided wrongly with no noise at all. More transmit power does not remove them.

2.5 Nyquist and the raised cosine

A pulse with zero interference

A pulse adds nothing to the samples of its neighbours when it is zero at every sampling instant but its own. This holds however long the pulse lasts.

ZERO INTERFERENCE

$$p(kT_b) = \begin{cases} 1, & k = 0 \\ 0, & k \neq 0 \end{cases}$$

The sinc pulse has this property. Here $\text{sinc}(x) = \sin(\pi x)/(\pi x)$, with $\text{sinc}(0) = 1$.

THE SINC PULSE

$$p(t) = \text{sinc}\left(\frac{t}{T_b}\right), \qquad p(kT_b) = \frac{\sin(\pi k)}{\pi k} = 0 \quad (k \neq 0)$$

For example, with $T_b = 0.1$ ms, $p(t) = 0$ every 0.1 ms except at $t = 0$.

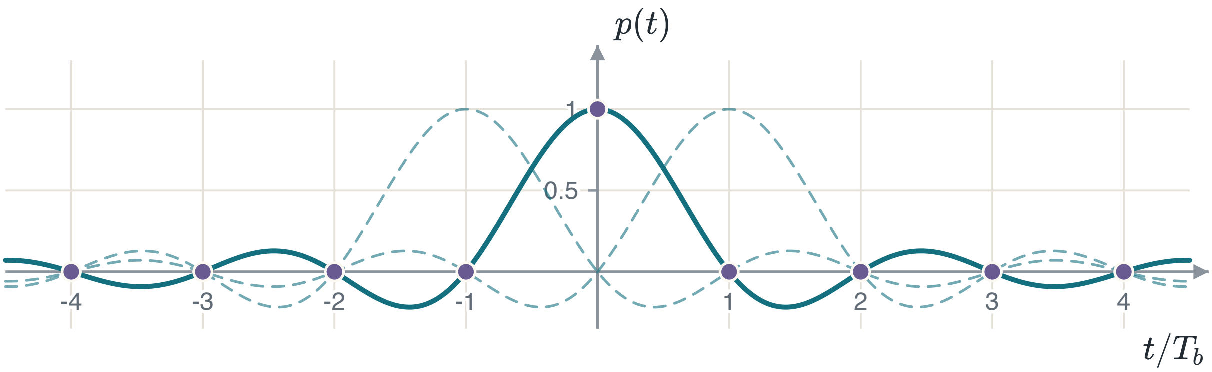


FIGURE 2.29 $p(t) = \text{sinc}(t/T_b)$ (solid) with two of its neighbours (dashed). At every sampling instant only one pulse is non-zero.

Nyquist's criterion for zero intersymbol interference

The zero-interference condition can be written in frequency. Sample the pulse at $t = kT_b$ and use the replication result of Chapter 1, with $f_s = R_b = 1/T_b$.

SAMPLE THE PULSE

$$p_\delta(t) = \sum_k p(kT_b)\,\delta(t - kT_b)$$

$$P_\delta(f) = R_b \sum_n P(f - nR_b)$$

Zero interference leaves only the sample $p(0) = 1$. Then $p_\delta(t) = \delta(t)$, whose transform is 1. Set $P_\delta(f) = 1$ and divide by R_b.

NYQUIST'S CRITERION

$$p(kT_b) = \delta[k] \iff \sum_{n=-\infty}^{\infty} P(f - nR_b) = T_b$$

The copies of $P(f)$ spaced R_b apart must add to a constant.

For example, the triangle $P(f) = T_b(1 - |f|/R_b)$ for $|f| < R_b$ has the pulse $p(t) = \mathrm{sinc}^2(t/T_b)$. Its copies add to T_b, so the criterion holds. Directly, $p(T_b) = \mathrm{sinc}^2(1) = 0$.

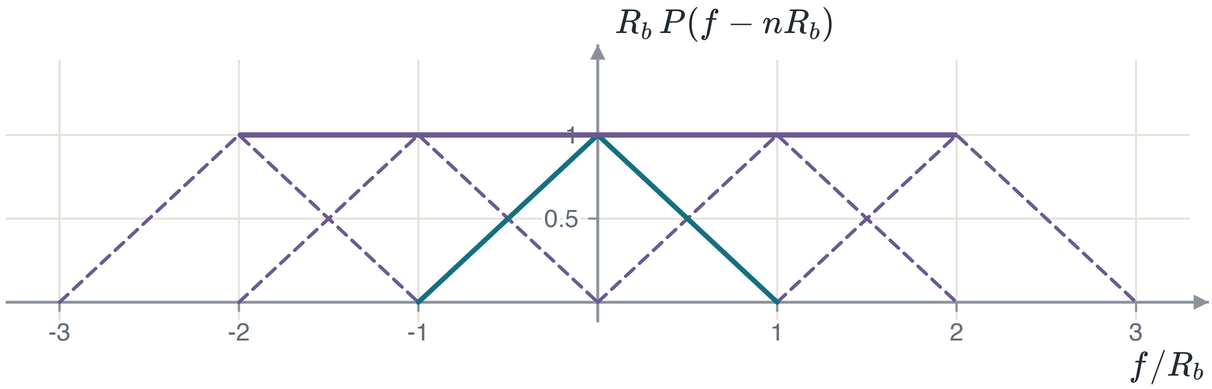


FIGURE 2.30 The triangular $P(f)$ of width $2R_b$ (cyan) and its copies at $\pm R_b$ and $\pm 2R_b$ (dashed). Their sum (violet) is a constant, so $p(t)$ has zero interference.

The minimum bandwidth

The copies are spaced R_b apart. If $P(f) = 0$ for $|f| > B_T$ with $B_T < R_b/2$, the copies leave gaps, and their sum cannot be constant.

NYQUIST BANDWIDTH

$$B_T \geq W = \frac{R_b}{2}$$

$W = R_b/2$ is the **Nyquist bandwidth**. For example, a link at $R_b = 64$ kb/s needs at least $W = 64/2 = 32$ kHz.

At $B_T = W$ only one spectrum fills the axis: the rectangle of width $2W = R_b$. Its inverse transform is the sinc pulse.

THE ONLY PULSE AT $B_T = W$

$$P(f) = \frac{1}{2W}\,\Pi\left(\frac{f}{2W}\right) \leftrightarrow p(t) = \mathrm{sinc}(2Wt) = \mathrm{sinc}\left(\frac{t}{T_b}\right)$$

$$p(t) = \int_{-W}^{W} \frac{1}{2W} e^{j2\pi f t}\, df = \frac{\sin(2\pi W t)}{2\pi W t} = \mathrm{sinc}(2Wt)$$

$\Pi(f/2W)$ is 1 for $|f| < W$ and 0 elsewhere. Since $2W = R_b = 1/T_b$, this pulse is the sinc pulse above.

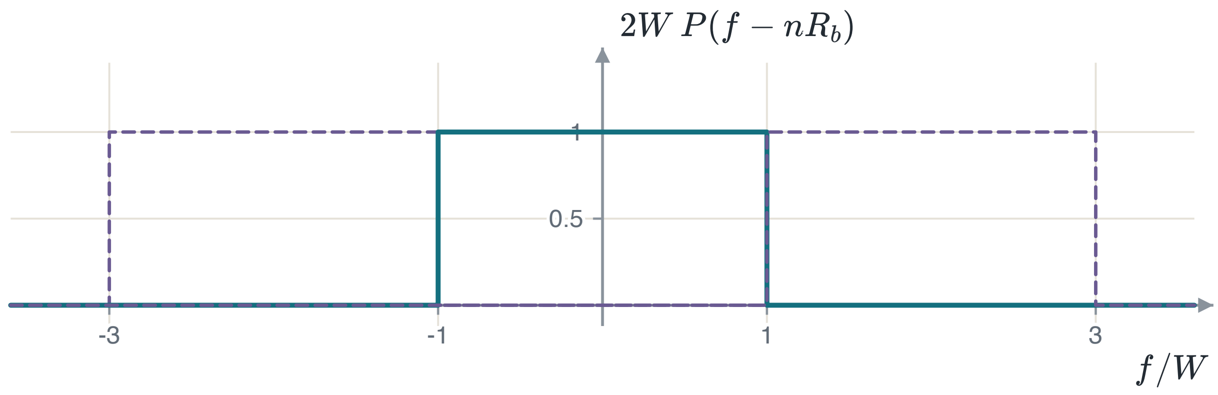


FIGURE 2.31 The rectangle of width $2W = R_b$ (cyan) and its copies at multiples of R_b (dashed). They meet edge to edge and fill the axis with no gap.

Timing and the sinc pulse

The sinc pulse is impractical. Suppose the receiver samples at $t = \epsilon$ instead of $t = 0$. Every neighbour then adds a small term.

A LATE SAMPLE

$$y_0 = a_0 \operatorname{sinc}\left(\frac{\epsilon}{T_b}\right) + \sum_{k\neq 0} a_k \operatorname{sinc}\left(k + \frac{\epsilon}{T_b}\right)$$

The wanted term barely drops. At $\epsilon = 0.1T_b$ it is $\operatorname{sinc}(0.1) = \sin(0.1\pi)/(0.1\pi) = 0.309/0.314 = 0.984.$ The interference is the problem.

⚠ SLOW TAILS

The terms fall only as $1/(\pi|k|)$. The sum of their sizes behaves like $\sum 1/k$, which grows without bound. A sinc pulse also lasts forever in both directions.

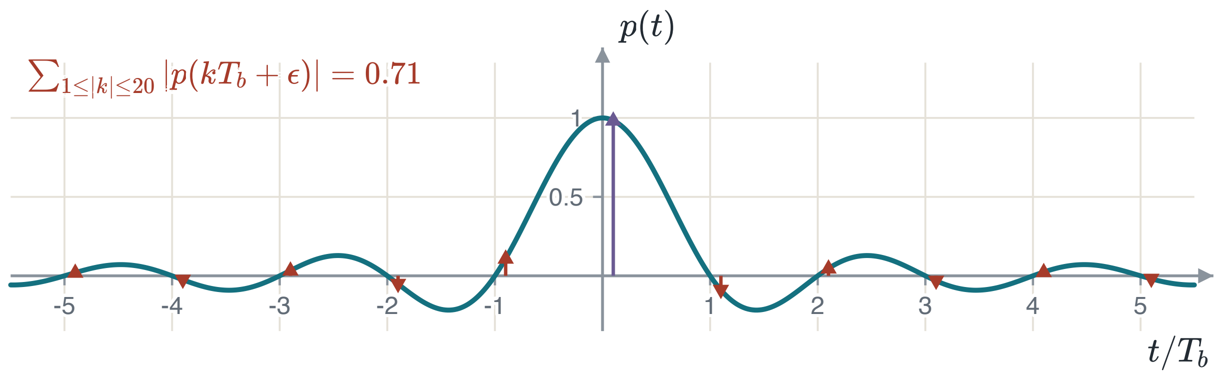


FIGURE 2.32 A sample taken $\epsilon = 0.1T_b$ late. The violet stem is the wanted term and each red stem is a neighbour's contribution. The sum of their sizes over $1 \leq |k| \leq 20$ is 0.71.

The raised-cosine spectrum

A practical pulse widens the spectrum beyond W and tapers its edges. The **raised cosine** does this and still meets Nyquist's criterion. Its **roll-off factor** α runs from 0 to 1.

RAISED-COSINE SPECTRUM

$$P(f) = \begin{cases} \dfrac{1}{2W}, & |f| < f_1 \\ \dfrac{1}{4W}\left[1 + \cos\dfrac{\pi(|f| - f_1)}{2W - 2f_1}\right], & f_1 \le |f| < 2W - f_1 \\ 0, & |f| \ge 2W - f_1 \end{cases}$$

$W = R_b/2$ and $f_1 = W(1 - \alpha)$. The spectrum is flat up to f_1, then falls along half a cosine period to zero at $2W - f_1$.

BANDWIDTH

$$B_T = 2W - f_1 = 2W - W(1 - \alpha) = W(1 + \alpha) = \frac{R_b}{2}(1 + \alpha)$$

$\alpha = 0$ is the rectangle, with $B_T = W$. $\alpha = 1$ doubles the bandwidth to R_b. For example, $W = 5$ kHz and $\alpha = 0.25$ give $B_T = 5(1.25) = 6.25$ kHz.

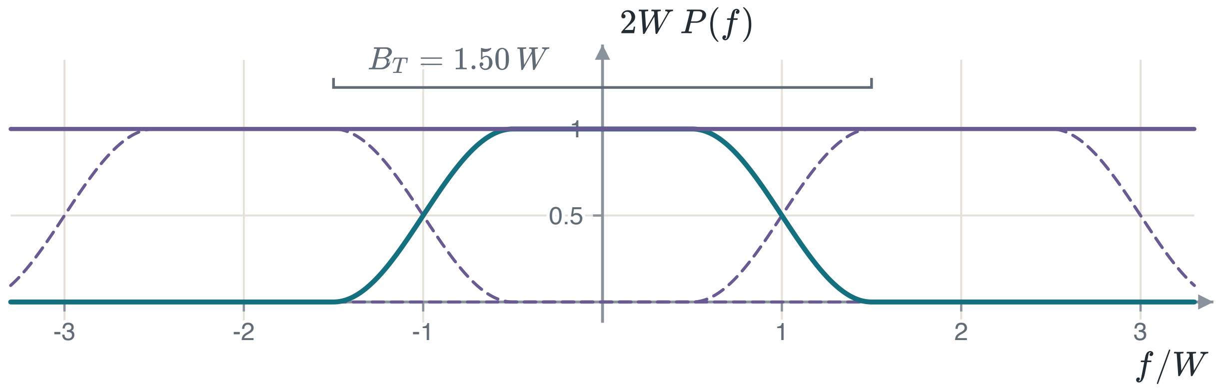


FIGURE 2.33 The raised-cosine spectrum with $\alpha = 0.5$ (cyan), its copies at $\pm 2W$ (dashed) and their sum (violet). The spectrum widens from W to $B_T = 1.5W$, and the copies still add to a constant.

The raised-cosine pulse

The inverse transform of the raised-cosine spectrum is the raised-cosine pulse.

THE PULSE

$$p(t) = \operatorname{sinc}\left(\frac{t}{T_b}\right)\frac{\cos(\pi\alpha t/T_b)}{1 - 4\alpha^2 t^2/T_b^2}$$

The sinc factor keeps the zeros at $t = kT_b$. Here $\operatorname{sinc}(x) = \sin(\pi x)/(\pi x)$.

FASTER DECAY

For $\alpha > 0$ the second factor falls as $1/t^2$, so the tails fall as $1/|t|^3$. A timing error then collects little interference.

The second factor can add zeros of its own. With $\alpha = 1$, $\cos(1.5\pi) = 0$ and the denominator $1 - 4(2.25) = -8$ is not zero, so $p(1.5T_b) = 0$.

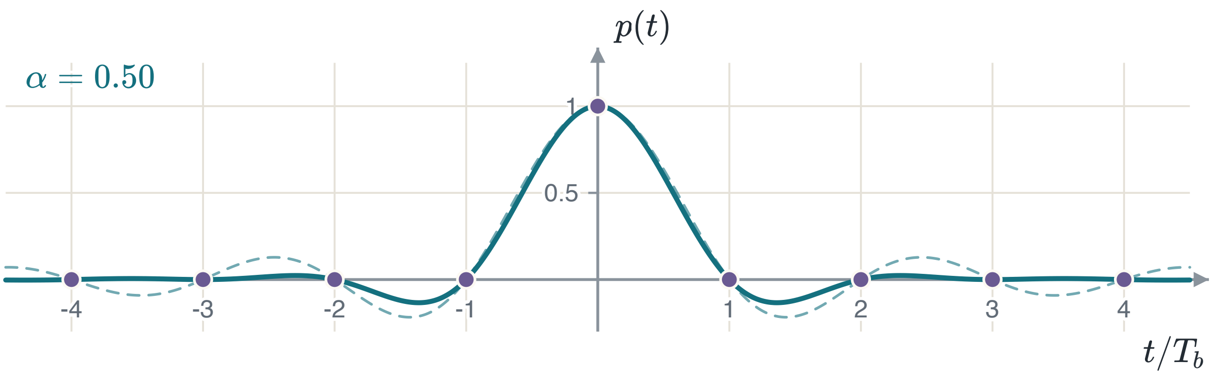


FIGURE 2.34 The raised-cosine pulse with $\alpha = 0.5$ (solid) against the sinc pulse (dashed). The zeros at $t = kT_b$ stay where they are, and the tails die out sooner.

EXAMPLE 2.5 — A RAISED-COSINE LINK

GIVEN A link sends $R_b = 20$ kb/s with raised-cosine pulses and $\alpha = 0.5$.

FIND W, f_1 and B_T.

METHOD Each of the three follows from R_b and α: $W = R_b/2$, $f_1 = W(1-\alpha)$ and $B_T = W(1+\alpha)$. The spectrum is flat up to f_1 and falls to zero at B_T, with its half-height point at W.

SOLUTION $W = 20/2 = 10$ kHz. $f_1 = 10(1-0.5) = 5$ kHz. $B_T = 10(1+0.5) = 15$ kHz.

CHECK $B_T - W = W - f_1 = 5$ kHz, so the roll-off is symmetric about W.

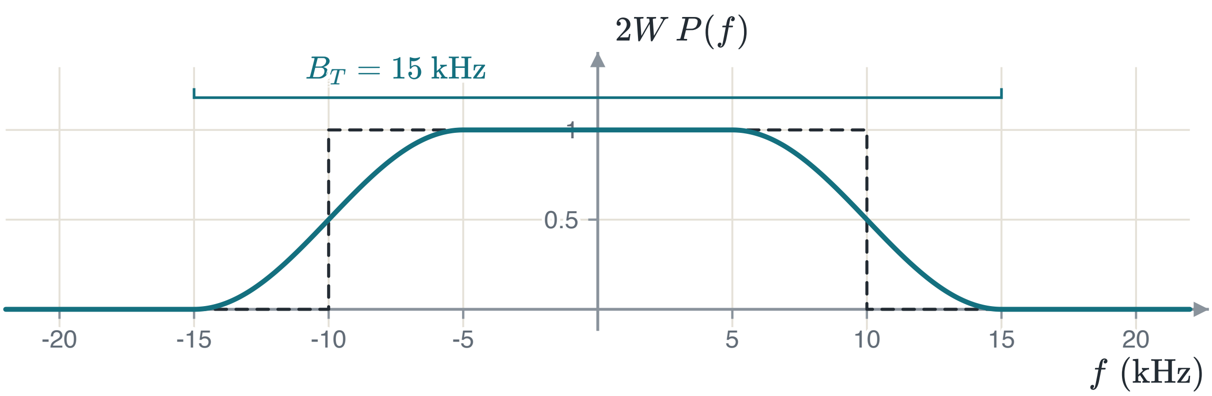


FIGURE 2.35 Example 2.5: the ideal rectangle of width $2W$ (dashed) and the raised-cosine spectrum with $\alpha = 0.5$, which reaches $B_T = 15$ kHz.

COMMON ERROR

Use $W = R_b/2 = 10$ kHz, not $R_b = 20$ kHz. The wrong W gives $B_T = 30$ kHz, twice the true value.

Splitting the raised cosine

The raised cosine can be shared between the two ends of the link. H_T shapes the pulse at the transmitter and H_R filters at the receiver. Their product must meet Nyquist's criterion.

SPLIT

$$H_T(f)\,H_R(f) = P(f), \qquad H_T(f) = H_R(f) = \sqrt{P(f)}$$

MATCHED AND FREE OF INTERFERENCE

$H_R = H_T^*$ for a real, even $\sqrt{P(f)}$, so the receive filter is matched to the pulse. The link then has both the smallest P_b and zero interference.

At $f = W$ the raised cosine is at half height, $2W\,P(W) = 0.5$. With $\alpha = 0.5$, the square-root filter there is $\sqrt{2W\,P(W)} = \sqrt{0.5} = 0.707$.

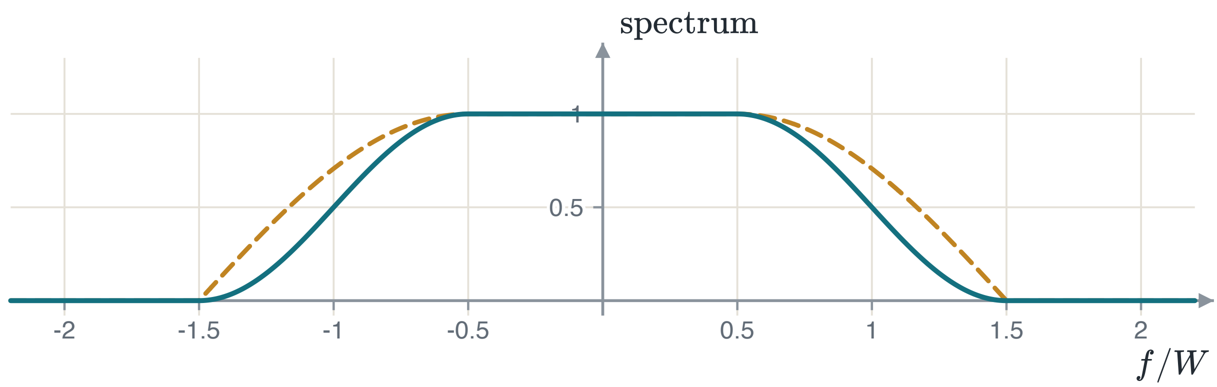


FIGURE 2.36 The raised cosine with $\alpha = 0.5$ (cyan) and its square root (amber, dashed). Two square-root filters in a row give the raised cosine.

Pulse shaping around us

Broadcast and mobile standards fix the roll-off factor and the bandwidth.

- **Satellite television.** Satellite television at $R_s = 27.5$ Mbaud with $\alpha = 0.35$ uses $B_T = 13.75(1.35) = 18.6$ MHz at baseband.
- **3G mobile.** 3G mobile sends 3.84 Mchip/s with $\alpha = 0.22$. $B_T = 1.92(1.22) = 2.34$ MHz fits inside the ± 2.5 MHz of its 5 MHz channel.
- **The pulse.** The raised-cosine pulse of $\alpha = 0.35$ is zero at every other sampling instant, so the samples carry no interference.
- **Timing.** A sample taken ϵ late collects interference, summed here over $1 \leq |k| \leq 30$. The raised cosine with $\alpha = 0.5$ collects far less than the sinc pulse.

ROLL-OFF IN STANDARDS

Satellite television uses $\alpha = 0.35$, 0.25 or 0.20. 3G mobile uses $\alpha = 0.22$. A small α saves bandwidth.

BANDWIDTH

$B_T = \frac{R_s}{2}(1 + \alpha)$ at baseband. On a carrier the occupied band is twice that, $R_s(1 + \alpha)$.

⚠ TIMING

A small α gives long tails. A sample taken late then collects interference from many neighbours.

2.6 The link in practice

The equalizer

An **equalizer** is a filter after the channel that removes interference. The simplest one adds w times the waveform one bit earlier.

ONE TAP

$$z(t) = y(t) + w\,y(t - T_b)$$

The pulse after the equalizer is $r(t) + w\,r(t - T_b)$. Sample it m bits after its own bit. The pulse leaves its own tail, and the delayed copy leaves the tail one bit younger.

THE EQUALIZED PULSE

$$\begin{aligned} g_0 &= 1 - q \\ g_m &= (1-q)\,q^m + w\,(1-q)\,q^{m-1} \\ &= (1-q)\,q^{m-1}(q + w), \qquad m \geq 1 \end{aligned}$$

ZERO FORCING

Each tail sample is q times the one before, so $w = -q$ cancels them all. The tap also adds a delayed copy of the noise.

For example, $BT_b = 0.25$ gives $q = 0.208$. The tap $w = -0.208$ makes $q + w = 0$, so every g_m with $m \geq 1$ vanishes.

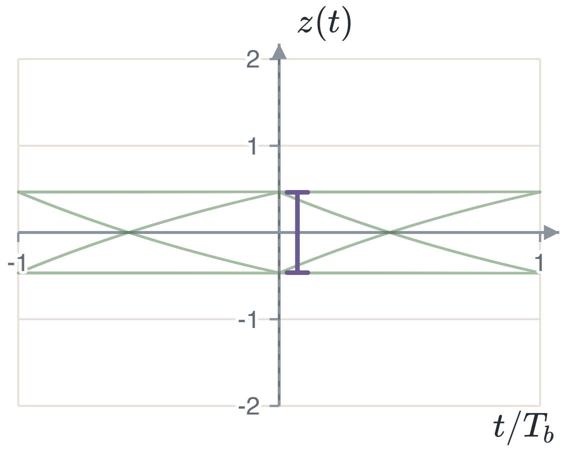


FIGURE 2.37 The eye after a one-tap equalizer, for $BT_b = 0.1$ and $w = -0.53$. Without the tap this eye is closed.

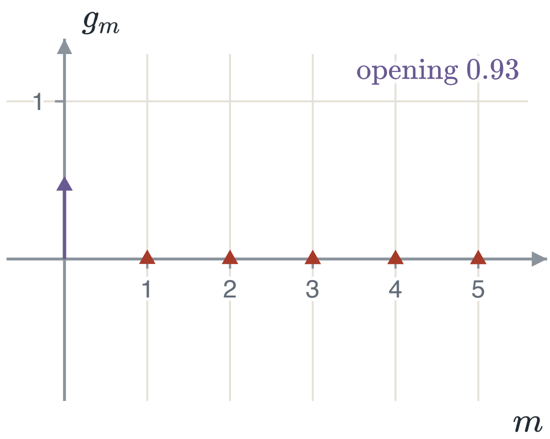


FIGURE 2.38 The equalized pulse sampled m bits later. Violet is the bit itself and red is what it leaves on later bits. With w near $-q = -0.533$ the tail is almost gone.

Finding the sampling instant

The receiver must find the sampling instant on its own. The matched-filter output peaks at the instant it wants. An **early–late gate** finds that peak.

The gate takes two extra samples, δ before and δ after the clock τ. The output is symmetric about its peak, so the two agree only there.

> ERROR SIGNAL
>
> $$e = |y(\tau - \delta)| - |y(\tau + \delta)|$$
>
> If $e > 0$ the clock is late and moves earlier. If $e < 0$ it moves later.

The larger sample lies on the side of the peak. For example, an early sample of 0.62 and a late sample of 0.80 put the peak after the clock, so the clock is early.

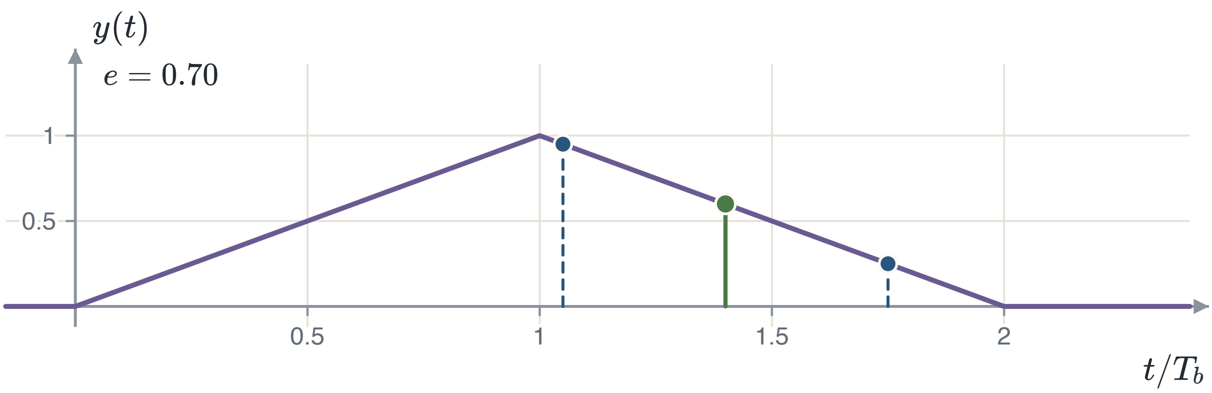


FIGURE 2.39 The early–late gate on the matched-filter output of a rectangular pulse, with the clock (green) $0.4T_b$ late. The early and late samples (dashed) sit $\delta = 0.35T_b$ either side. Here $e = 0.70 > 0$, so the clock moves earlier.

The spectrum of a bit stream

A bit stream occupies a band of frequencies, and a neighbouring channel must stay clear of it. For independent, equally likely ± 1 bits the spectrum has a simple form.

POWER SPECTRUM

$$S(f) = \frac{|G_T(f)|^2}{T_b}$$

$G_T(f)$ is the transmit filter. The stream has the spectrum of one pulse.

⚠ NEIGHBOURS

A rectangular pulse keeps only 90% of its power inside $|f - f_c| < R_b$. A raised cosine stops at $(1+\alpha)R_b/2$.

On a carrier each band is $(1+\alpha)R_b$ wide. For example, channels 1.25 MHz apart that carry $R_b = 1$ Mb/s each need $(1+\alpha)(1\ \mathrm{MHz}) \leq 1.25$ MHz, so $\alpha \leq 0.25$.

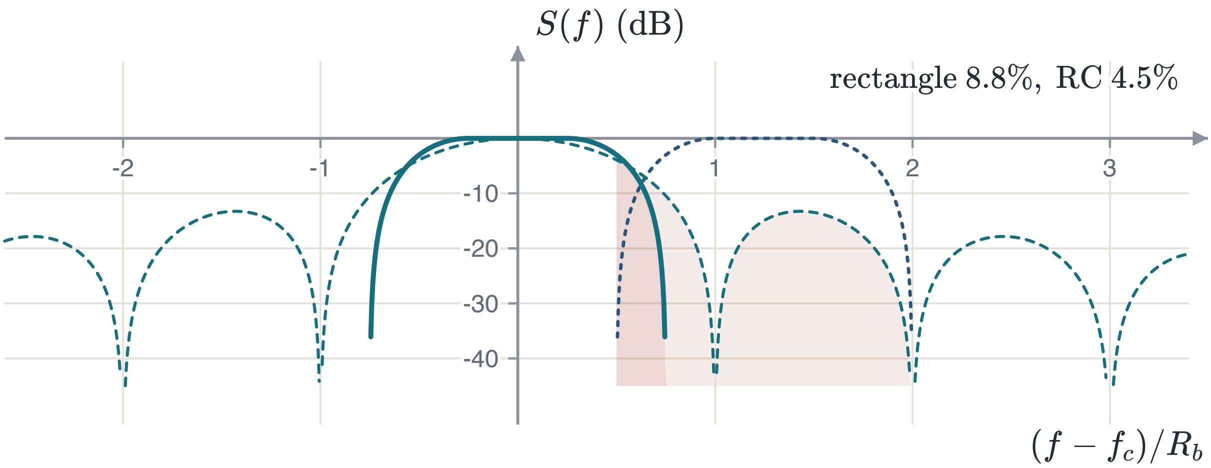


FIGURE 2.40 Spectra around one carrier, for $\alpha = 0.5$ and a channel spacing $\Delta = 1.25 R_b$. The raised cosine is solid, the rectangular pulse dashed, and the neighbouring channel dotted. The red area is power that lands in the neighbour, and the numbers give its share for each pulse.

Regenerative repeaters

A long link is split into hops, with a repeater after each hop. There are two kinds of repeater.

- An **analog** repeater amplifies the signal and the noise together.
- A **regenerative** repeater decides the bits and sends a clean waveform on.

After K hops the two chains differ. A regenerative chain adds up the small error probability of each hop. An analog chain adds up the noise of each hop, so its N_0 grows K times.

AFTER K HOPS

$$P_{\text{regen}} \approx K\,Q\big(\sqrt{2E_b/N_0}\big)$$
$$P_{\text{analog}} = Q\big(\sqrt{2E_b/(KN_0)}\big)$$

For $K = 50$ hops and $P_b = 10^{-6}$, regeneration needs $E_b/N_0 = 11.8$ dB and amplification 27.5 dB. A chain of 100 regenerative repeaters with $p = 10^{-6}$ a hop has $P_b \approx Kp = 100 \times 10^{-6} = 10^{-4}$.

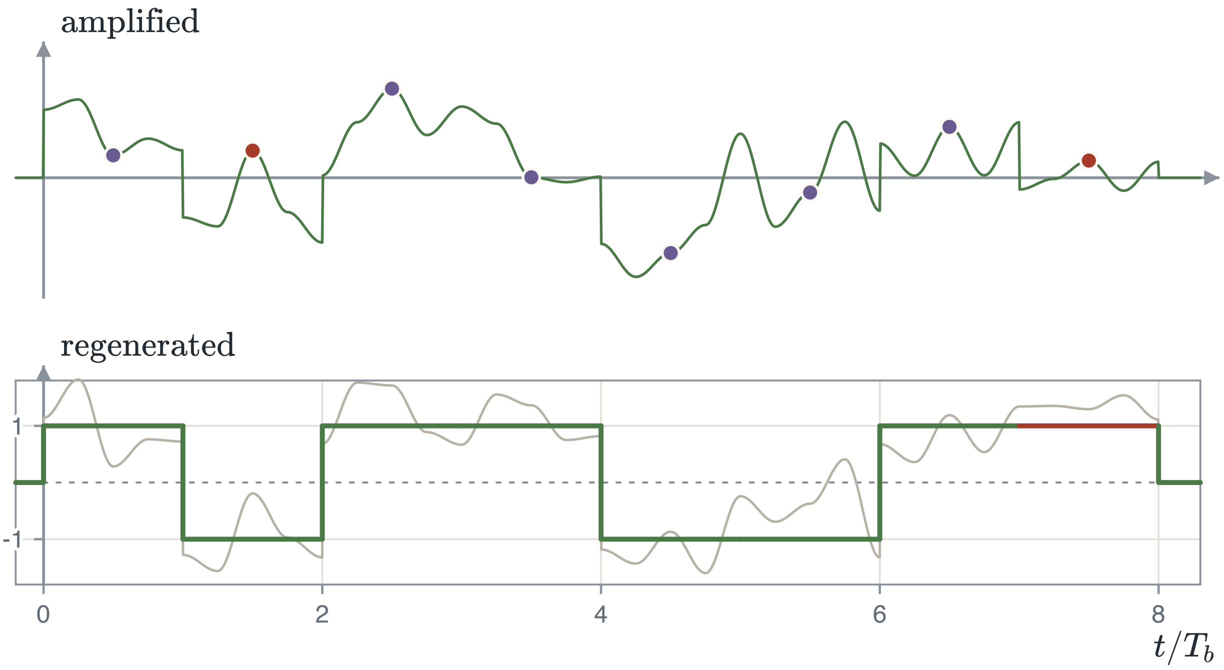


FIGURE 2.41 Four hops of the bits $1, 0, 1, 1, 0, 0, 1, 0$. Top: the amplified chain carries the noise of every hop, and a red dot is a wrong decision. Bottom: the regenerative chain sends a clean waveform on, and a wrong bit, drawn red, stays wrong.

2.7 Summary

From bits to decisions

The whole link has four stages.

1. Send each bit as $\pm\sqrt{E_b}\,\psi(t)$.
2. The channel adds white noise of density $N_0/2$.
3. Correlate with $\psi(t)$ and read y_k at the end of the bit.
4. Decide by comparing y_k with λ_{opt}.

TWO LIMITS

Noise sets $P_b = Q\big(\sqrt{2E_b/N_0}\big)$. Bandwidth sets $R_b \le 2B_T/(1+\alpha)$ for zero interference.

For example, a polar link at $E_b/N_0 = 9.6$ dB with equal priors has $E_b/N_0 = 10^{0.96} = 9.12$. Then $P_b = Q(\sqrt{18.2}) = Q(4.27) = 9.7 \times 10^{-6}$, about 10^{-5}.

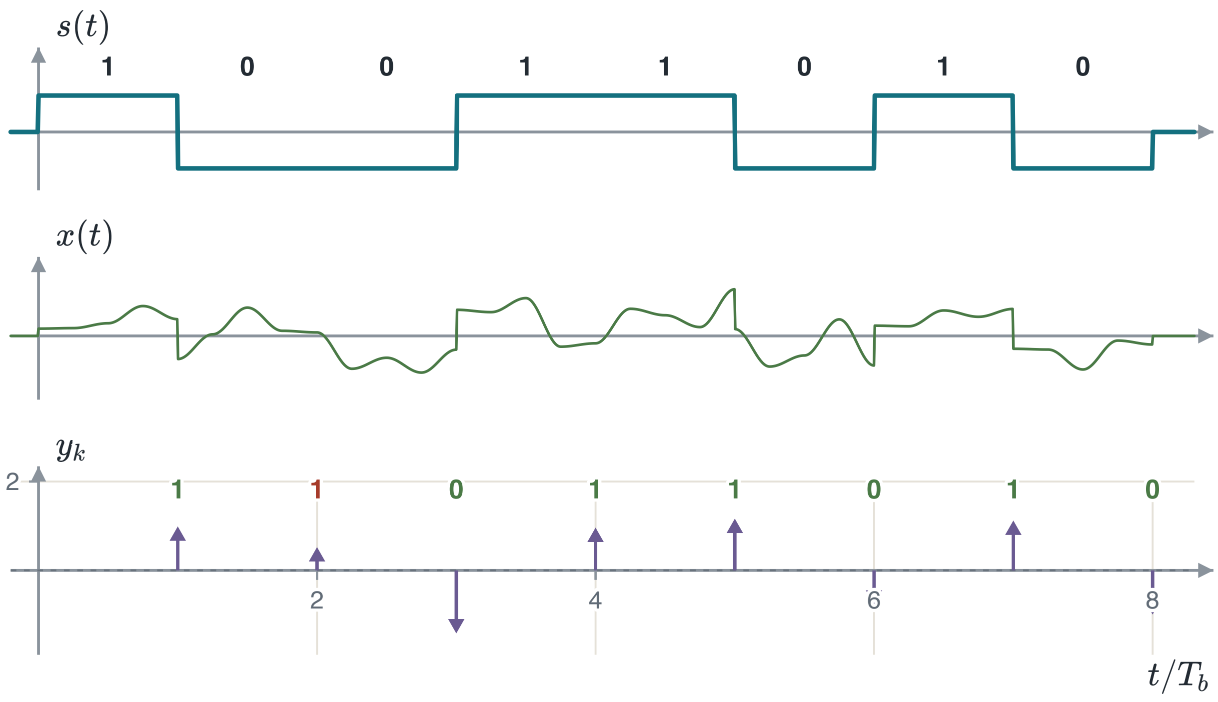


FIGURE 2.42 Eight bits through the link. Each bit becomes $\pm A$ in $s(t)$, noise is added in $x(t)$, and the correlator gives one number y_k a bit. The sign of y_k is the decision, and the second decision is wrong (red).

Quick check

Six short predictions, each with its reason.

1 **Matched filter.** The filter matched to $s(t)$ on $[0, T]$ has the impulse response $s(t)$, $s(T-t)$ or $s(t-T)$?

Answer: $s(T-t)$. Reverse the pulse in time, then shift it right by T.

2 **Peak SNR.** With $E = 2\ \mu$J and $N_0 = 10^{-7}$ W/Hz, the largest peak SNR is 20, 40 or 80?

Answer: 40, since $2E/N_0 = 2(2\times10^{-6})/10^{-7} = 40$.

3 **Threshold.** With equal priors, the optimal threshold is 0, $N_0/4$ or $\sqrt{E_b}$?

Answer: 0, since $\ln(p_0/p_1) = \ln 1 = 0$.

4 **Error probability.** E_b/N_0 rises from 4 to 8. P_b falls from 2.3×10^{-3} to 1.2×10^{-3}, 3.2×10^{-5} or 10^{-9}?

Answer: 3.2×10^{-5}, since $Q(\sqrt{16}) = Q(4) = 3.2\times10^{-5}$.

5 **Interference.** An RC channel has $BT_b = 0.25$. The worst-case total interference q is 0.208, 0.5 or 0.792?

Answer: 0.208, since $q = e^{-2\pi(0.25)} = e^{-\pi/2} = 0.208$.

6 **Bandwidth.** $R_b = 1$ Mb/s with a raised cosine of $\alpha = 0.5$ needs $B_T = 0.5$, 0.75 or 1 MHz?

Answer: 0.75 MHz, since $B_T = \frac{R_b}{2}(1+\alpha) = 0.5(1.5) = 0.75$ MHz.

Results of the chapter

Twelve questions recall the results this chapter carries forward.

TABLE 2.1 Summary of Chapter 2: baseband transmission of digital signals.

QUESTION	RESULT	ANCHOR
Which filter gives the largest peak SNR?	The matched filter $h(t) = k\,g(T-t)$, or $H(f) = k\,G^*(f)\,e^{-j2\pi fT}$, sampled at $t = T$.	PS CH8.3.2
What is that largest peak SNR?	$\eta_{\max} = 2E/N_0$. It depends on the pulse energy E, not on its shape.	PS CH8.3.2
Where do the two symbols of polar signalling sit?	At $\pm\sqrt{E_b}$ on the axis of the unit-energy $\psi(t)$, a distance $2\sqrt{E_b}$ apart.	PS CH8.2.1
When do the correlator and the matched filter agree?	At $t = T_b$, where both give $\int_0^{T_b} x(t)\,\psi(t)\,dt$, for any shape of ψ.	PS CH8.3.1
What noise is left in the demodulator output?	$y = s_m + n$, with n Gaussian of mean 0 and variance $N_0/2$.	PS CH8.3.1
Where does the optimal threshold go?	$\lambda_{\text{opt}} = \frac{N_0}{4\sqrt{E_b}} \ln \frac{p_0}{p_1}$, at 0 for equal priors and toward the less likely symbol otherwise.	PS CH8.3.3
What is $Q(x)$?	The Gaussian tail $Q(x) = \frac{1}{2}\,\mathrm{erfc}(x/\sqrt{2})$, the chance that a unit Gaussian exceeds x.	PS CH8.3.3
What is P_b for polar signalling?	$P_b = Q\left(\sqrt{2E_b/N_0}\right)$ with equal priors and a matched filter.	PS CH8.3.3
How much interference does an RC channel add?	A bit m bits back adds $(1-q)q^m$, with $q = e^{-2\pi BT_b}$. The worst case totals q.	PS CH10.1.1
When does the eye close?	The opening is $2(1-2q)$. It closes at $BT_b = \ln 2/(2\pi) = 0.110$, and errors then occur without noise.	—
What is Nyquist's criterion?	$p(kT_b) = \delta[k]$ exactly when $\sum_n P(f - nR_b) = T_b$. It needs $B_T \geq R_b/2$.	PS CH10.3.1
What bandwidth does a raised cosine need?	$B_T = \frac{R_b}{2}(1+\alpha)$. The tails fall as $1/\lvert t\rvert^3$ for $\alpha > 0$.	PS CH10.3.1

Match the filter to the pulse and read it at the end of the bit. Place the threshold from the priors. Check B_T against $R_b/2$ before choosing a pulse. Chapter 3 extends these steps to more than two waveforms.

Projects to try

Four optional projects use the chapter on simulated links. Each gives an aim, what it practises, a few steps and what to look for.

ⓘ MEASURE A BIT ERROR RATE

Simulate a polar link and compare the measured error rate with $Q\left(\sqrt{2E_b/N_0}\right)$.
Practises: the correlator receiver and its decision statistic. Setting the noise variance from E_b/N_0. How many bits a reliable estimate of P_b needs.

1. Draw 10^6 random bits and map them to ± 1.
2. Add Gaussian noise of variance $N_0/2$ for E_b/N_0 from 0 to 10 dB.
3. Decide by the sign of each value and count the errors.
4. Plot the measured P_b on a logarithmic axis with the Q curve over it.

Look for: the points lie on the curve until the error count gets small. There they scatter, because too few errors were seen.

ⓘ FIND AN ECHO IN NOISE

Hide a known pulse in strong noise and find its delay with a matched filter.
Practises: correlation with a template. Why a long pulse with a wide bandwidth gives a sharp peak. Reading a delay from the peak position.

1. Make a chirp of 1000 samples that sweeps a wide band.
2. Place it at an unknown delay inside $10\,000$ samples of noise that hides it.
3. Correlate the record with the chirp and find the largest peak.
4. Repeat with a plain rectangular pulse of the same energy.

Look for: the chirp gives one narrow peak at the true delay. The rectangle gives a broad triangle whose top is hard to place in noise.

ⓘ DRAW THE EYE OF A CABLE

Send random bits through a lowpass channel and watch the eye close as the bandwidth falls.
Practises: intersymbol interference from a bandlimited channel. The eye diagram as an overlay of short pieces. The link between eye opening and error rate.

1. Build an RC lowpass with time constant $\tau = 1/(2\pi B)$.
2. Pass 2000 random polar bits through it at $BT_b = 0.5$, 0.25 and 0.12.
3. Cut the output into pieces $2T_b$ long around each sample and plot them together.
4. Add a little noise and count the errors at each bandwidth.

Look for: the eye narrows as BT_b falls and is nearly shut near 0.11. The error count jumps once the eye closes.

SHAPE PULSES WITH A RAISED COSINE

Compare the sinc pulse with raised-cosine pulses in bandwidth and in sensitivity to timing.

Practises: the raised-cosine pulse and its roll-off α. The spectrum of a pulse train. Why a small α needs a precise clock.

1. Build raised-cosine pulses with $\alpha = 0$, 0.25 and 0.5, truncated to $\pm 8T_b$.
2. Send random polar bits with each and plot the spectrum of the result.
3. Sample the received train slightly late, by $0.05T_b$ to $0.3T_b$.
4. Plot the eye for each α at each timing error.

Look for: a larger α widens the spectrum. It also keeps the eye open for a larger timing error.

CHAPTER

3

Geometric representation of signal waveforms

Each waveform of a signal set becomes a point. Energies, distances and the receiver are then read off a picture.

The chapter has two results. First, an orthonormal basis turns each waveform into N numbers. Energy is squared length, and the energy of a difference is squared distance.

Second, Gram–Schmidt finds a basis for any set, with $N \leq M$. Sets with the same constellation need the same receiver.

3.1 Signals as vectors

Two waveforms, two axes

Take two waveforms $s_0(t)$ and $s_1(t)$ that take the values ± 1 on $[0, T]$. The figure draws both and their product.

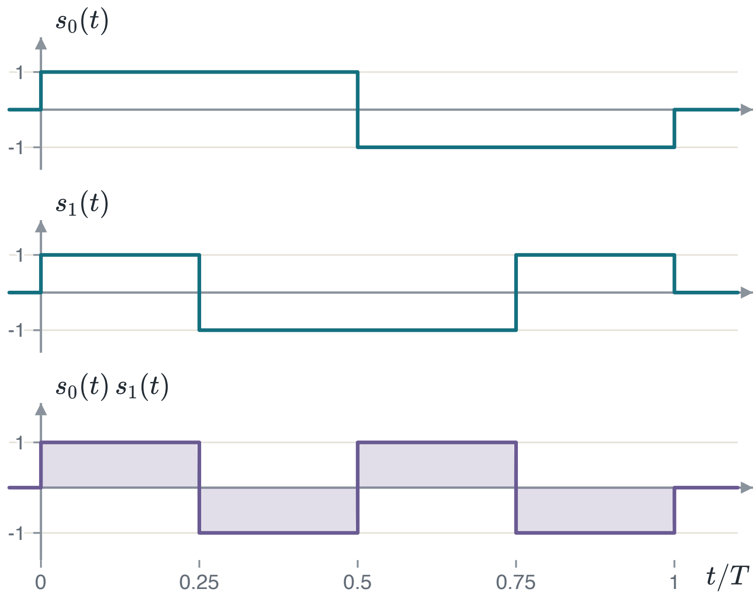


FIGURE 3.1 $s_0(t)$ and $s_1(t)$ take the values ± 1 on $[0, T]$. Their product is $+1$ on two quarters and -1 on the other two.

Neither waveform is a multiple of the other. No single $\psi(t)$ writes both as $s_m \psi(t)$, so the one-number receiver of Chapter 2 does not apply.

The receiver computes one number for each waveform, with two correlators.

TWO CORRELATORS

$$y_0 = \int_0^T x(t)\, s_0(t)\, dt, \qquad y_1 = \int_0^T x(t)\, s_1(t)\, dt$$

The pair (y_0, y_1) is a point in a plane.

Take $T = 1$ and integrate the product $s_0(t)\, s_1(t)$. It is $+1$ on two quarters and -1 on the other two. So the two areas are 0.5 each, and they cancel.

THE PRODUCT INTEGRATES TO ZERO

$$\int_0^1 s_0(t)\, s_1(t)\, dt = 0.5 - 0.5 = 0$$

An orthonormal basis

A vector written against the unit axes $\mathbf{e}_1$ and $\mathbf{e}_2$ is the list (a_1, a_2). Its length and its angle to another vector come from that list.

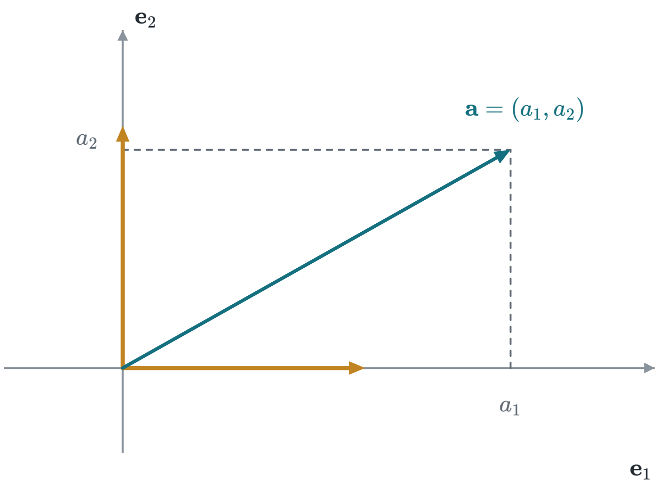


FIGURE 3.2 A vector written against the unit axes $\mathbf{e}_1$ and $\mathbf{e}_2$ is the list (a_1, a_2). Its length and its angle to another vector come from that list.

The inner product multiplies matching entries and adds them. The length is the root of the inner product with itself.

VECTORS

$$\langle \mathbf{a}, \mathbf{b} \rangle = \sum_{k=1}^{N} a_k b_k, \qquad \|\mathbf{a}\| = \sqrt{\langle \mathbf{a}, \mathbf{a} \rangle}$$

For signals the sum becomes an integral.

SIGNALS

$$\langle x, y\rangle = \int_{-\infty}^{\infty} x(t)\, y(t)\, dt, \qquad \|x\|^2 = \int_{-\infty}^{\infty} x^2(t)\, dt = E_x$$

The squared length of a signal is its energy.

The set $\{\psi_1, \ldots, \psi_N\}$ is **orthonormal** when the integral of $\psi_j(t)\psi_k(t)$ is 1 for $j = k$ and 0 for $j \neq k$.

ORTHONORMAL SET

$$\int \psi_j(t)\, \psi_k(t)\, dt = \begin{cases} 1, & j = k \\ 0, & j \neq k \end{cases}$$

Each function has unit energy, and each pair is orthogonal.

For example, take $\psi(t) = c$ on $[0, 2]$ and zero elsewhere. Its energy is $\int_0^2 c^2\, dt = 2c^2$. Unit energy needs $2c^2 = 1$, so $c = 1/\sqrt{2} = 0.707$.

The coordinates of a waveform

One integral for each axis takes the waveform apart into N numbers. This step is called **analysis**.

ANALYSIS

$$s_{ij} = \int_0^T s_i(t)\, \psi_j(t)\, dt, \qquad j = 1, \ldots, N$$

The N numbers build the waveform back. This step is called **synthesis**.

SYNTHESIS

$$s_i(t) = \sum_{j=1}^{N} s_{ij}\, \psi_j(t), \qquad \mathbf{s}_i = (s_{i1}, \ldots, s_{iN})$$

The list $\mathbf{s}_i$ is the **signal vector**.

The figure shows synthesis on two unit pulses. The waveform $s_1\psi_1(t) + s_2\psi_2(t)$ and the point (s_1, s_2) change together.

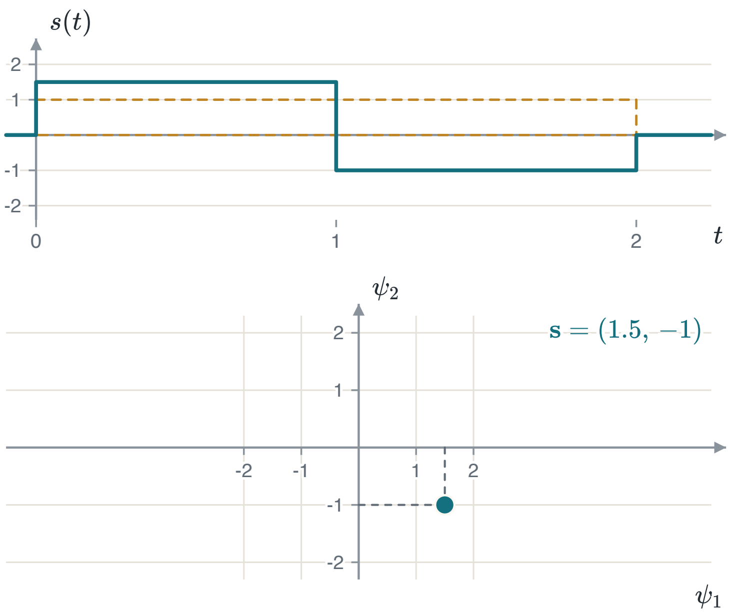


FIGURE 3.3 The waveform $s_1\psi_1(t) + s_2\psi_2(t)$ and the point (s_1, s_2), drawn for $s_1 = 1.5$ and $s_2 = -1$. The dashed pulses are ψ_1 and ψ_2.

For example, take $\psi_1 = 1$ on $[0, 1)$ and $\psi_2 = 1$ on $[1, 2)$. Let $s(t) = 3$ on $[0, 1)$ and -1 on $[1, 2)$. Each coordinate is one analysis integral.

THE COORDINATES OF ONE WAVEFORM

$$s_1 = \int_0^1 3 \cdot 1 \, dt = 3$$

$$s_2 = \int_1^2 (-1) \cdot 1 \, dt = -1$$

So $\mathbf{s} = (3, -1)$.

The analyzer and the synthesizer

The **analyzer** multiplies the waveform by each $\psi_j(t)$ and integrates over $[0, T]$. It is a bank of correlators, one for each basis function.

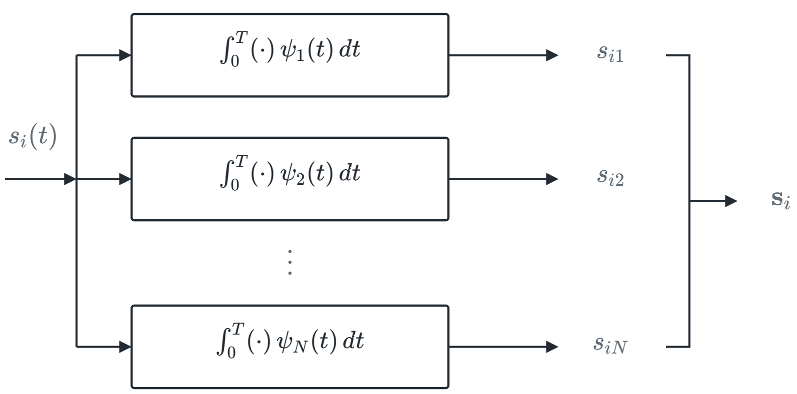


FIGURE 3.4 The analyzer: one correlator for each basis function. Its N outputs are the coordinates, and together they form the signal vector $\mathbf{s}_i$.

The receiver of Chapter 4 is this bank of correlators.

The **synthesizer** scales each $\psi_j(t)$ by s_{ij} and adds. The transmitter can build every waveform of the set from N basis functions.

For example, a set of $M = 8$ waveforms spans $N = 2$ dimensions. The analyzer needs one correlator for each basis function, so it needs $N = 2$ correlators. The number of waveforms that share the two axes does not matter.

Inner products are preserved

An integral over two waveforms equals a sum over N pairs of coordinates.

KEY RESULT: INNER PRODUCTS

$$\int_{-\infty}^{\infty} x(t)\, y(t)\, dt = \sum_{k=1}^{N} x_k y_k = \langle \mathbf{x}, \mathbf{y} \rangle$$

The proof takes three moves. Write $x = \sum_j x_j \psi_j$ and $y = \sum_k y_k \psi_k$. Move the integral inside both sums. Then use $\int \psi_j \psi_k \, dt = 1$ for $j = k$ and 0 otherwise.

PROOF IN THREE MOVES

$$\begin{aligned} \int_{-\infty}^{\infty} x(t)\, y(t)\, dt &= \int_{-\infty}^{\infty} \Big(\sum_{j=1}^{N} x_j \psi_j(t)\Big)\Big(\sum_{k=1}^{N} y_k \psi_k(t)\Big) dt \\ &= \sum_{j=1}^{N} \sum_{k=1}^{N} x_j y_k \int_{-\infty}^{\infty} \psi_j(t)\, \psi_k(t)\, dt \\ &= \sum_{k=1}^{N} x_k y_k = \langle \mathbf{x}, \mathbf{y} \rangle \end{aligned}$$

In the double sum, each term with $j \neq k$ is 0 and each term with $j = k$ keeps $x_k y_k$.

For example, take $x = 2\psi_1 + \psi_2$ and $y = \psi_1 - \psi_2$ on the two unit pulses. So $\mathbf{x} = (2, 1)$ and $\mathbf{y} = (1, -1)$. The product $x(t)\, y(t)$ is 2 on $[0, 1)$ and -1 on $[1, 2)$.

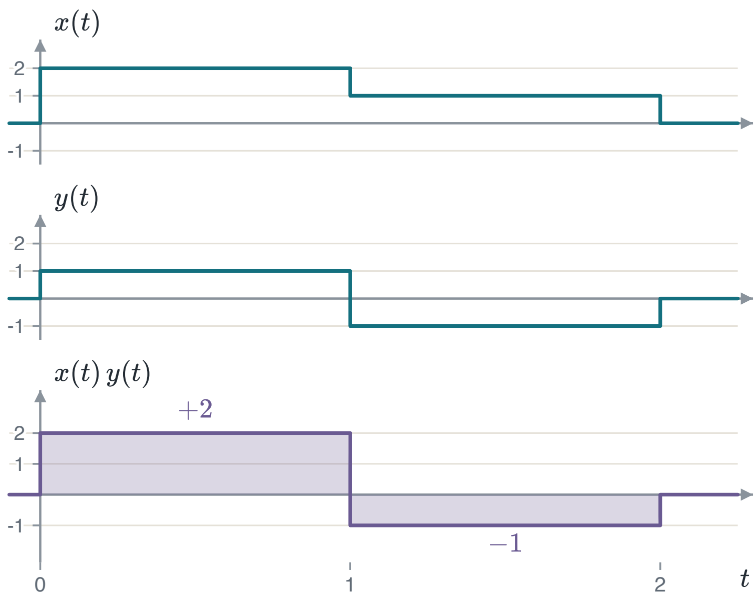


FIGURE 3.5 $x = 2\psi_1 + \psi_2$ and $y = \psi_1 - \psi_2$ on the two unit pulses, with their product and its signed area.

THE SIGNED AREA AND THE COORDINATES AGREE

$$\int_0^2 x(t)\,y(t)\,dt = 2 - 1 = 1, \qquad x_1 y_1 + x_2 y_2 = 2(1) + 1(-1) = 1$$

Energy and distance

Put $y = x$ in the key result. Energy is the squared distance from the point to the origin.

ENERGY

$$E_i = \int_0^T s_i^2(t)\,dt = \sum_{j=1}^{N} s_{ij}^2 = \|\mathbf{s}_i\|^2$$

Apply the key result to the difference $s_i - s_k$, whose vector is $\mathbf{s}_i - \mathbf{s}_k$.

DISTANCE

$$d_{ik}^2 = \|\mathbf{s}_i - \mathbf{s}_k\|^2 = \int_0^T \big(s_i(t) - s_k(t)\big)^2 dt$$

In the figure, $\mathbf{s}_1$ and $\mathbf{s}_2$ stay on the circle of radius $\sqrt{2}$. So both energies stay 2, and only their distance d changes with the angle between them.

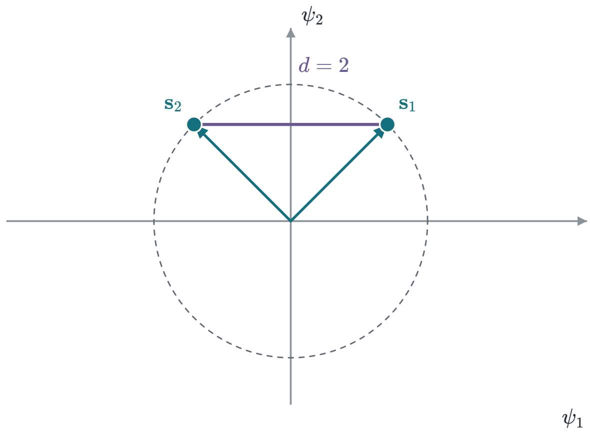


FIGURE 3.6 $\mathbf{s}_1$ and $\mathbf{s}_2$ on the circle of radius $\sqrt{2}$, drawn with the angle between them at $90°$. Both energies are 2. Only their distance d changes with the angle.

> ⚠ DISTANCE, NOT ENERGY
>
> Noise moves the received point. Two points far apart are hard to confuse, so d decides the error rate. Chapter 4 derives the rule.

For example, $\mathbf{s}_1 = (1, 1)$ and $\mathbf{s}_2 = (1, -1)$ give $\mathbf{s}_1 - \mathbf{s}_2 = (0, 2)$. So $d_{12} = \sqrt{0^2 + 2^2} = 2$. Its square, 4, is the energy of $s_1 - s_2$.

Example 3.1 finds a basis for four signals by reading their pieces. The figure shows the four signals and their points.

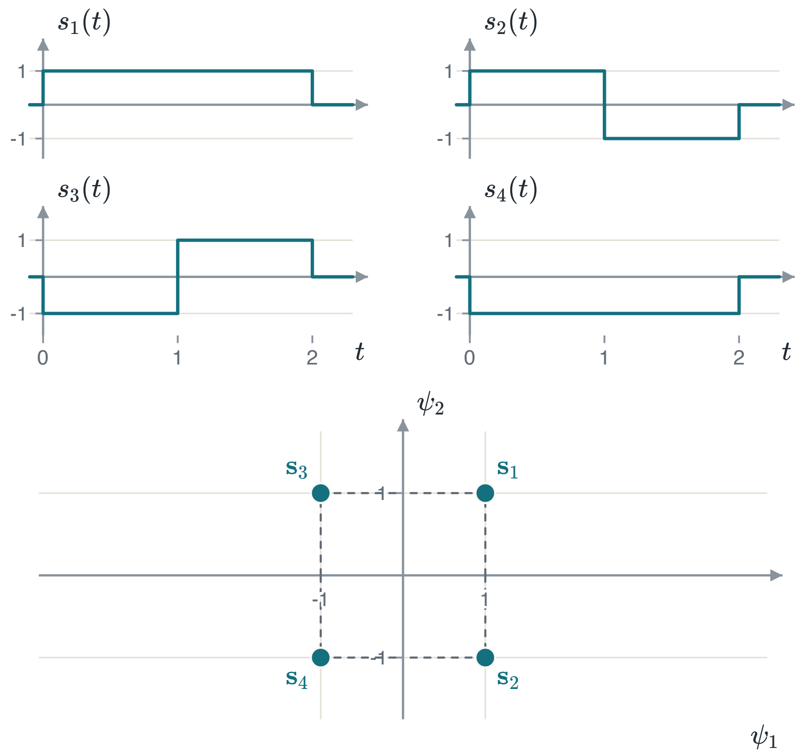


FIGURE 3.7 The four signals of Example 3.1, each constant on $[0, 1)$ and on $[1, 2)$, and their four points.

EXAMPLE 3.1 — A BASIS BY INSPECTION

GIVEN Four signals, each ± 1 on $[0, 1)$ and on $[1, 2)$, giving all four sign pairs.

FIND — A basis and the four vectors. How many basis functions?

METHOD — Every signal is built from a pulse on $[0, 1)$ and a pulse on $[1, 2)$, so two basis functions are enough. Read off the pieces, then check unit energy and orthogonality.

SOLUTION — Take $\psi_1(t) = 1$ for $0 \le t < 1$ and $\psi_2(t) = 1$ for $1 \le t < 2$. Then $\int \psi_1^2\, dt = \int \psi_2^2\, dt = 1$. Also $\int \psi_1 \psi_2\, dt = 0$, because the pulses never overlap. Each coordinate is the height on that half times its width 1. So $\mathbf{s}_1 = (1, 1)$, $\mathbf{s}_2 = (1, -1)$, $\mathbf{s}_3 = (-1, 1)$ and $\mathbf{s}_4 = (-1, -1)$, with $N = 2$.

CHECK — The four points form a square. Each energy is $1^2 + 1^2 = 2$ from the vectors, and $\int_0^2 (\pm 1)^2\, dt = 2$ from the waveforms.

⊗ COMMON ERROR

Use $\psi_1 = s_1/\sqrt{E_1}$ as an axis, not s_1 itself. Here $E_1 = 2$, so an unscaled axis puts every coordinate a factor $\sqrt{2}$ off.

Orthogonal signals around us

Four everyday systems use orthogonal signal sets.

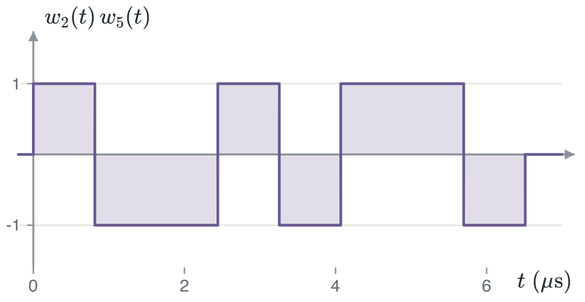


FIGURE 3.8 IS-95 phones share one band, each call with its own Walsh code at 1.2288 Mchip/s. Two codes multiply to $+1$ on four chips and -1 on four, so $\sum_n w_2[n]\, w_5[n] = 0$.

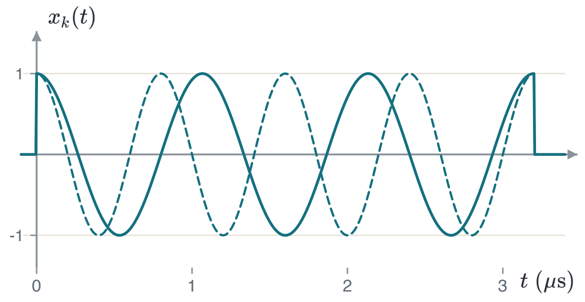


FIGURE 3.9 Wi-Fi spaces its subcarriers $\Delta f = 312.5$ kHz apart. Over one symbol of $T = 1/\Delta f = 3.2$ μs, $\cos(2\pi k \Delta f t)$ for $k = 3$ and $k = 4$ are orthogonal.

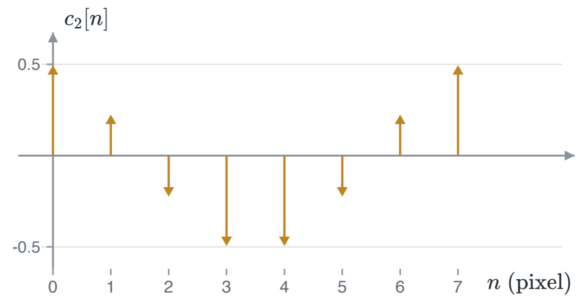


FIGURE 3.10 JPEG writes each row of 8 pixels against the DCT basis $c_k[n] = \frac{1}{2}\cos\big(\pi(2n+1)k/16\big)$, $k \ge 1$. Here $k = 2$. The basis is orthonormal, so a row's energy is the sum of its squared coefficients.

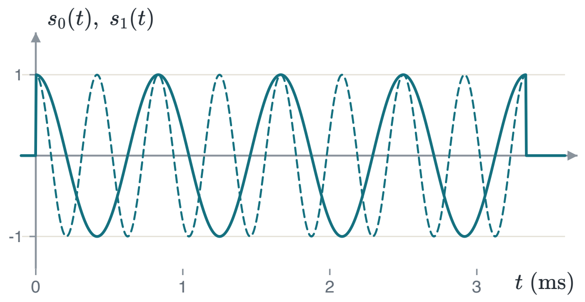


FIGURE 3.11 The Kansas City standard stored bits on cassette tape at 300 baud: 4 cycles of 1200 Hz for a 0, 8 cycles of 2400 Hz for a 1. Whole cycles in $T = 3.33$ ms make the tones orthogonal.

Each system picks signals whose inner product is zero. A correlator matched to one of them gives zero for all the others.

Two sinusoids are orthogonal over T when each fits a whole number of cycles in T and the numbers differ.

> ⚠ THE INTERVAL
>
> Orthogonality holds over the exact interval. A receiver that integrates over a shifted window lets the signals leak into each other.

3.2 Constellations

The constellation diagram

The **constellation diagram** is the signal vectors drawn in the space of their basis. It has one point for each waveform and one axis for each basis function.

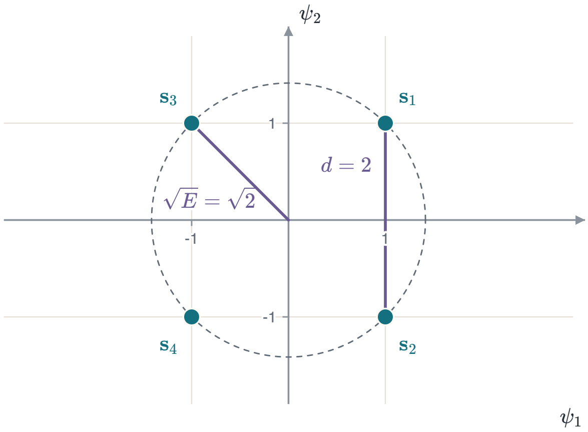


FIGURE 3.12 The four signals of Example 3.1 as a constellation, with an energy read as a radius and a distance between two points.

Three things are read off the constellation.

1. Distance to the origin: $\sqrt{E_i}$.
2. Distance between two points: the root of the energy of their difference.
3. Number of axes: the correlators the receiver needs.

For example, take the four points at $(\pm 1, \pm 1)$. Neighbours such as $(1, 1)$ and $(1, -1)$ differ by 2 in one coordinate, so $d_{\min} = 2$. The diagonal pairs are $2\sqrt{2} = 2.83$ apart.

Three binary signal sets as points

Two classic binary sets are compared by the distance between their two points. Each has average energy per bit E_b.

Antipodal signals satisfy $s_2 = -s_1$. They need one axis, and $d = 2\sqrt{E_b}$. Polar NRZ and BPSK are antipodal.

Orthogonal signals satisfy $\langle s_1, s_2 \rangle = 0$. They need two axes, and $d = \sqrt{2E_b}$. PPM and FSK are orthogonal.

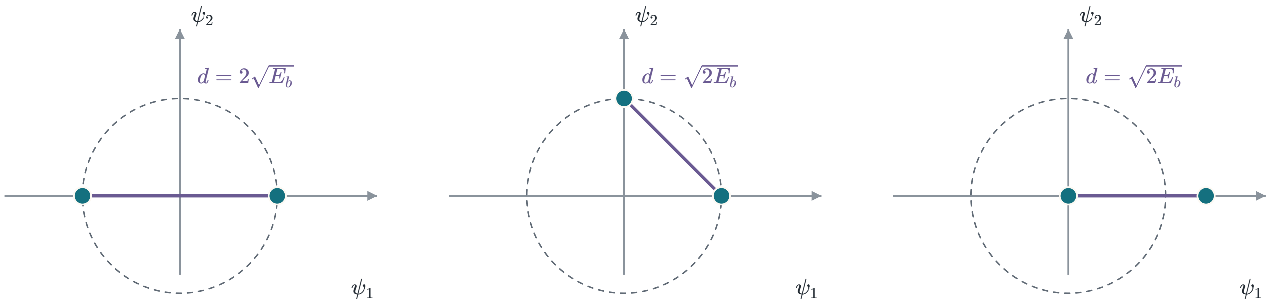


FIGURE 3.13 Three binary sets, each with average energy E_b, in units of $\sqrt{E_b}$: antipodal on the left, orthogonal in the middle and on-off on the right.

Find the energy each set needs for the same distance d. Square the two distance formulas and set them equal.

ENERGY FOR EQUAL DISTANCE

$$4E_b^{\text{anti}} = d^2 = 2E_b^{\text{orth}}$$

$$\frac{E_b^{\text{orth}}}{E_b^{\text{anti}}} = 2 = 3.01 \text{ dB}$$

Orthogonal signalling needs twice the energy for the same distance.

On-off keying sends 0 or $s(t)$ with equal priors and average energy E_b. The average is E_b, so $s(t)$ has energy $2E_b$. The points are 0 and $\sqrt{2E_b}$, so $d = \sqrt{2E_b}$.

A cosine and a sine as axes

A cosine and a sine of the same carrier form an orthonormal pair.

THE TWO AXES

$$\psi_1(t) = \sqrt{\tfrac{2}{T}}\cos(2\pi f_c t), \qquad \psi_2(t) = \sqrt{\tfrac{2}{T}}\sin(2\pi f_c t), \qquad 0 \le t \le T$$

$f_c T$ is a whole number, so each function holds whole cycles in $[0, T]$.

The factor $\sqrt{2/T}$ gives each axis unit energy. Use $2\cos^2 a = 1 + \cos 2a$, then integrate term by term.

UNIT ENERGY

$$\begin{aligned}\int_0^T \psi_1^2\, dt &= \frac{2}{T}\int_0^T \cos^2(2\pi f_c t)\, dt \\ &= \frac{1}{T}\int_0^T \big[1 + \cos(4\pi f_c t)\big]\, dt \\ &= 1 + \frac{\sin(4\pi f_c T)}{4\pi f_c T} = 1\end{aligned}$$

The sine is zero because $4\pi f_c T$ is a whole multiple of 2π. The same steps with $2\sin^2 a = 1 - \cos 2a$ give unit energy for ψ_2.

The two axes are orthogonal. Use $2\cos a \sin a = \sin 2a$.

ORTHOGONAL

$$\begin{aligned}\int_0^T \psi_1\psi_2\,dt &= \frac{2}{T}\int_0^T \cos(2\pi f_c t)\sin(2\pi f_c t)\,dt \\ &= \frac{1}{T}\int_0^T \sin(4\pi f_c t)\,dt \\ &= \frac{1-\cos(4\pi f_c T)}{4\pi f_c T} = 0\end{aligned}$$

The product holds $2f_cT$ whole cycles, and $\cos(4\pi f_c T) = 1$.

The waveform $\sqrt{2E/T}\cos(2\pi f_c t - \theta)$ sits at $(\sqrt{E}\cos\theta, \sqrt{E}\sin\theta)$. Expand the cosine of a difference to see it.

A CARRIER WITH PHASE θ

$$\begin{aligned}s(t) &= \sqrt{\tfrac{2E}{T}}\cos(2\pi f_c t - \theta) \\ &= \sqrt{\tfrac{2E}{T}}\left[\cos\theta\cos(2\pi f_c t) + \sin\theta\sin(2\pi f_c t)\right] \\ &= \sqrt{E}\cos\theta\,\psi_1(t) + \sqrt{E}\sin\theta\,\psi_2(t)\end{aligned}$$

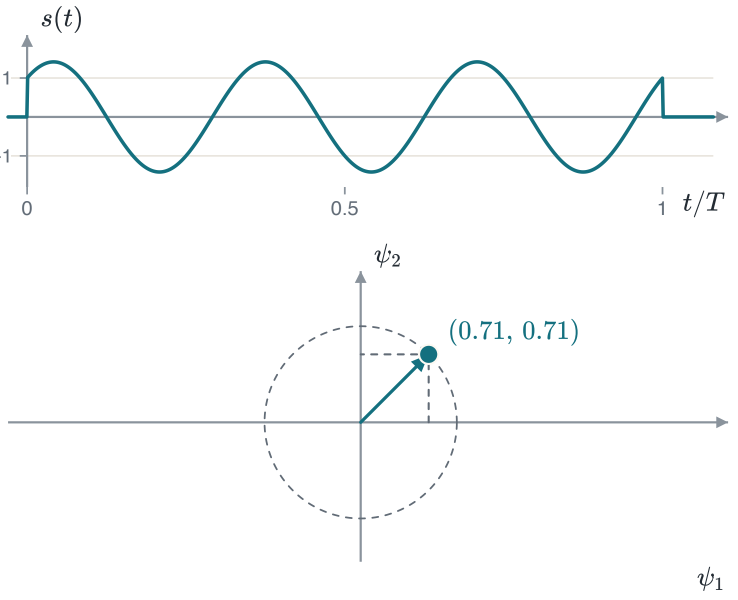


FIGURE 3.14 The waveform $\sqrt{2E/T}\cos(2\pi f_c t - \theta)$ and its point $(\sqrt{E}\cos\theta, \sqrt{E}\sin\theta)$, drawn for $\theta = 45°$, $E = 1$ and $f_cT = 3$.

For example, take $\theta = 90°$. Since $\cos(x - 90°) = \sin x$, $s(t) = \sqrt{E}\,\psi_2(t)$. Its point is $(0, \sqrt{E})$.

Example 3.2 finds the basis and the constellation of four carrier waveforms, and compares it with the pulse set of Example 3.1.

EXAMPLE 3.2 — FOUR CARRIER WAVEFORMS

GIVEN For $0 \le t \le T$ and whole $f_c T$: $s_{1,2}(t) = \mp\sqrt{2/T}\cos(2\pi f_c t) + \sqrt{2/T}\sin(2\pi f_c t)$ and $s_{3,4}(t) = \mp\sqrt{2/T}\cos(2\pi f_c t) - \sqrt{2/T}\sin(2\pi f_c t)$.

FIND The four points, and the energy of each.

METHOD By inspection, every waveform is already written in the cosine and sine basis. The coefficients are the coordinates: $s_i(t) = s_{i1}\psi_1(t) + s_{i2}\psi_2(t)$ with $s_{ij} = \pm 1$.

SOLUTION $\mathbf{s}_1 = (-1, 1)$, $\mathbf{s}_2 = (1, 1)$, $\mathbf{s}_3 = (-1, -1)$ and $\mathbf{s}_4 = (1, -1)$. Each vector has two entries ± 1, so each energy is $1 + 1 = 2$. The points form a square with $d_{\min} = 2$.

CHECK This is the square of the four pulse signals of Example 3.1. The waveforms differ, and the picture is the same.

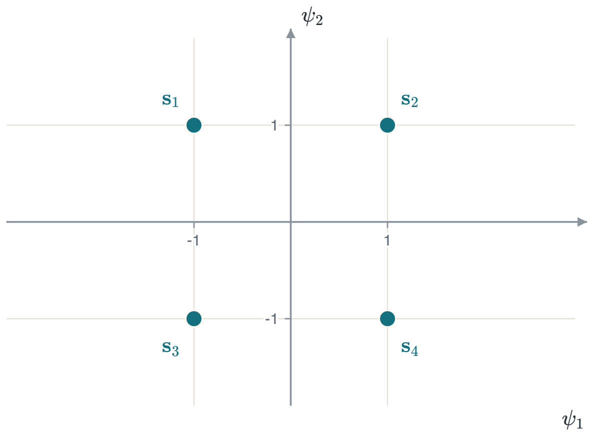


FIGURE 3.15 The four waveforms of Example 3.2 as points on the axes ψ_1 and ψ_2.

Geometric equivalence of signal sets

Different waveform sets can have the same constellation. They are called **geometrically equivalent**. They then need the same receiver and have the same error probability in white Gaussian noise.

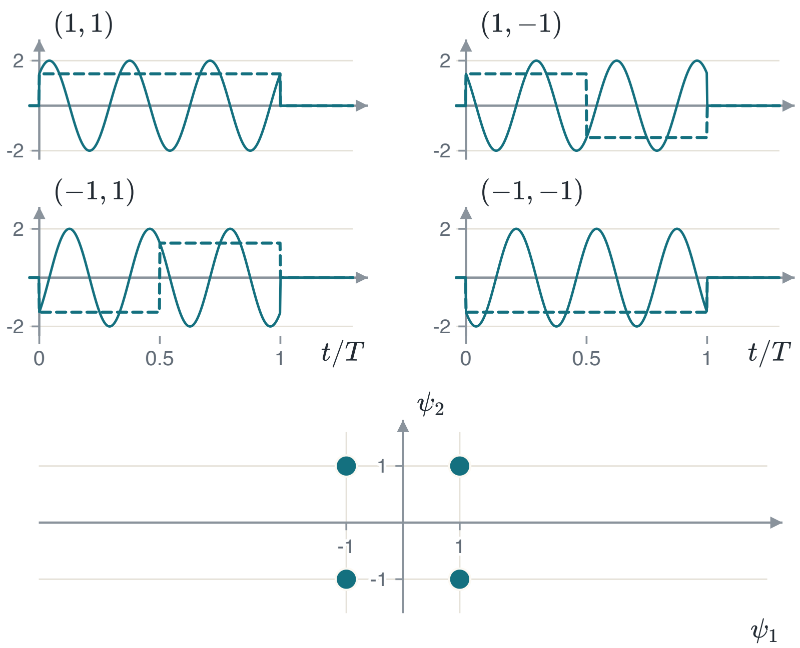


FIGURE 3.16 The pulse set and the carrier set. Each panel keeps its vector: its pair of pulses is dashed and its carrier waveform is solid. The constellation below is the same for both.

> ⚠ BANDWIDTH
>
> The constellation does not fix the bandwidth. The pulse set occupies frequencies near $f = 0$, and the carrier set occupies frequencies near f_c.

So two sets with the same constellation can differ in bandwidth. Error probability and receiver follow from the points. Bandwidth follows from the waveform shapes.

Points on a circle

M equal-energy carrier waveforms at equally spaced phases form the signal set below.

> THE SIGNAL SET
>
> $$s_i(t) = A\cos\left(2\pi f_c t - \tfrac{2\pi i}{M}\right), \quad i = 0, \dots, M-1$$
> $$\mathbf{s}_i = \sqrt{E}\left(\cos\tfrac{2\pi i}{M},\ \sin\tfrac{2\pi i}{M}\right), \quad E = \tfrac{A^2 T}{2}$$
>
> Every waveform has the same energy, so every point lies on one circle. The second line is the carrier result above with $\theta = 2\pi i/M$ and $A = \sqrt{2E/T}$.

Two neighbours are $2\pi/M$ apart in angle. The chord of that angle on a circle of radius $\sqrt{E}$ is $d_{\min}$. To find it, square the length of $\mathbf{s}_0 - \mathbf{s}_1$ and use $1 - \cos 2a = 2\sin^2 a$.

NEAREST NEIGHBOURS

$$\begin{aligned} d_{\min}^2 &= E\left[\left(1-\cos\tfrac{2\pi}{M}\right)^2+\sin^2\tfrac{2\pi}{M}\right] \\ &= E\left[2-2\cos\tfrac{2\pi}{M}\right] \\ &= 4E\sin^2\tfrac{\pi}{M} \end{aligned}$$

$$d_{\min} = 2\sqrt{E}\,\sin\frac{\pi}{M}$$

The middle step uses $\cos^2 a + \sin^2 a = 1$.

All M points share the circle of radius $\sqrt{E}$. Each doubling of M brings neighbours closer.

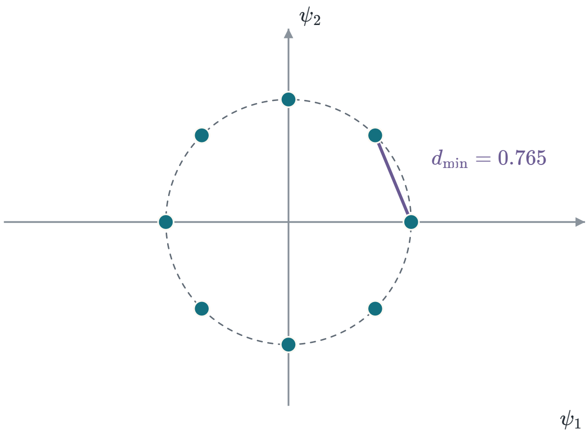


FIGURE 3.17 $M = 8$ points on the circle of radius $\sqrt{E}$, here $E = 1$, and the nearest pair.

For example, 8-PSK with $E = 1$ gives $d_{\min} = 2\sin(\pi/8) = 2(0.383) = 0.765$.

Constellations around us

Four real systems, each drawn at average energy 1.

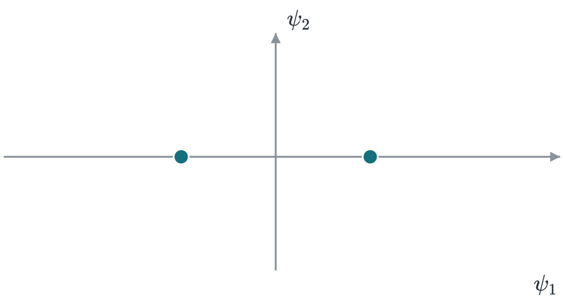


FIGURE 3.18 GPS satellites send their C/A code with BPSK on 1575.42 MHz. Two points, 1 bit a symbol, $d = 2\sqrt{E}$.

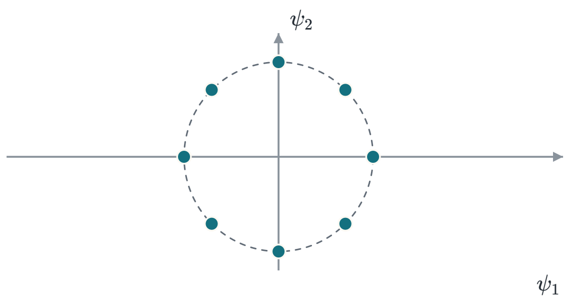


FIGURE 3.19 Satellite television (DVB-S2) can use 8PSK: 3 bits a symbol on one circle, with $d_{\min} = 2\sqrt{E}\sin(\pi/8) = 0.765\sqrt{E}$.

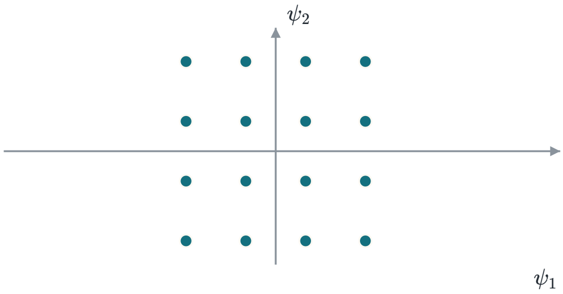


FIGURE 3.20 Wi-Fi at 36 Mb/s uses 16-QAM, 4 bits a symbol. Scaled to $E_{\text{avg}} = 1$, the grid step is $d_{\min} = 2/\sqrt{10} = 0.632$.

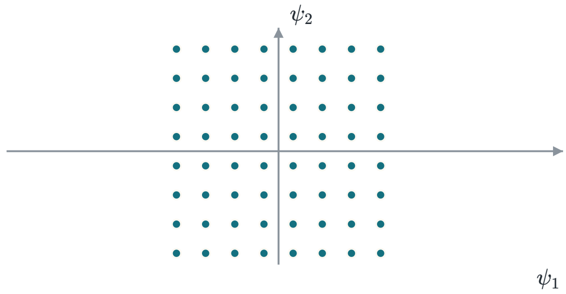


FIGURE 3.21 At 54 Mb/s Wi-Fi uses 64-QAM, 6 bits a symbol. With $E_{\text{avg}} = 1$, $d_{\min} = 2/\sqrt{42} = 0.309$.

M points carry $\log_2 M$ bits a symbol. Each doubling of M adds one bit.

All four are drawn with $E_{\text{avg}} = 1$. More points in the same energy must sit closer together.

⚠ SMALLER DISTANCE

$d_{\min}$ falls from 2 for BPSK to 0.309 for 64-QAM. Closer points need a cleaner channel.

3.3 The Gram–Schmidt procedure

The Gram–Schmidt steps

The **Gram–Schmidt procedure** finds an orthonormal basis for any set of signals. The first axis is the first signal scaled to unit energy.

STEP 1

$$E_1 = \int s_1^2(t)\, dt, \qquad \psi_1(t) = \frac{s_1(t)}{\sqrt{E_1}}$$

For each later signal, remove what the earlier axes already hold. Then scale the remainder to unit energy.

STEP k

$$s_{ki} = \int s_k(t)\, \psi_i(t)\, dt, \qquad g_k(t) = s_k(t) - \sum_{i=1}^{k-1} s_{ki}\, \psi_i(t)$$

$$\psi_k(t) = \frac{g_k(t)}{\sqrt{E_{g_k}}}, \qquad E_{g_k} = \int g_k^2(t)\, dt$$

The figure runs the procedure on two signals.

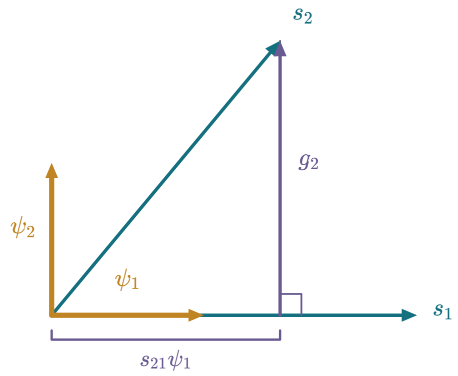


FIGURE 3.22 The procedure on two signals. The part of s_2 along ψ_1 is removed. The remainder g_2, scaled to unit length, is ψ_2.

> ⚠ STOP RULE
>
> If $g_k(t) = 0$, then s_k is a combination of earlier signals and adds no axis. So $N \le M$, with $N = M$ only when no signal is a combination of the others.

For example, let s_2 have energy 25 and $s_{21} = 3$. The remainder g_2 is perpendicular to ψ_1, so the squared lengths add: $E_2 = s_{21}^2 + E_{g_2}$.

> THE ENERGY OF THE REMAINDER
>
> $$E_{g_2} = E_2 - s_{21}^2 = 25 - 3^2 = 16$$

Example 3.3 runs the procedure on three pulses. It finds that two axes are enough.

EXAMPLE 3.3 — GRAM–SCHMIDT FOR THREE PULSES

GIVEN $s_1 = 1$ on $[0, 2)$, $s_2 = 1$ on $[2, 3)$ and $s_3 = 1$ on $[0, 3)$, each zero elsewhere.

FIND The basis and the vectors. How many axes?

METHOD Run Gram–Schmidt in the order s_1, s_2, s_3.

STEPS 1 AND 2 $E_1 = \int_0^2 1^2\, dt = 2$, so $\psi_1 = \frac{1}{\sqrt{2}}$ on $[0, 2)$. Then $s_{21} = \int s_2 \psi_1\, dt = 0$, because the two never overlap. So $g_2 = s_2$ and $\psi_2 = 1$ on $[2, 3)$.

STEP 3 ψ_1 is zero on $[2, 3)$, so $s_{31} = \int_0^2 1 \cdot \frac{1}{\sqrt{2}}\, dt = \sqrt{2}$. Also $s_{32} = \int_2^3 1 \cdot 1\, dt = 1$. Then $g_3 = s_3 - \sqrt{2}\,\psi_1 - \psi_2 = 0$.

SOLUTION $\mathbf{s}_1 = (\sqrt{2}, 0)$, $\mathbf{s}_2 = (0, 1)$ and $\mathbf{s}_3 = (\sqrt{2}, 1)$. Two axes are enough: $s_3 = s_1 + s_2$, so its remainder is zero and it adds no axis.

CHECK Energies from the vectors are 2, 1 and 3. They are the same as $\int s_i^2\, dt$: height 1 over lengths 2, 1 and 3.

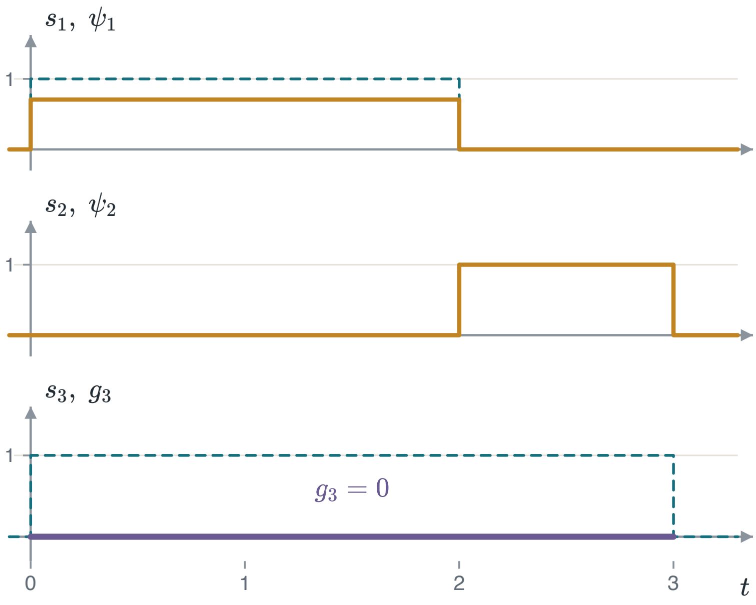


FIGURE 3.23 Each row holds one signal of Example 3.3, dashed, with the basis function it produces. The third leaves nothing: $g_3 = 0$.

Example 3.4 runs the procedure on four signals that span three dimensions.

EXAMPLE 3.4 – FOUR SIGNALS, THREE AXES

GIVEN $s_1 = 1$ on $[0, 2)$. $s_2 = 1$ on $[0, 1)$ and -1 on $[1, 2)$. $s_3 = -1$ on $[0, 1)$ and 1 on $[1, 3)$. $s_4 = 1$ on $[0, 3)$. Each is zero elsewhere.

FIND How many dimensions does the set span?

METHOD Run Gram–Schmidt in the order s_1, s_2, s_3, s_4. Each coordinate is a sum of piece heights times piece widths.

STEPS 1 AND 2 $E_1 = \int_0^2 1^2\,dt = 2$, so $\psi_1 = s_1/\sqrt{2}$. Then $s_{21} = \frac{1}{\sqrt{2}}(1 - 1) = 0$, so $g_2 = s_2$, with energy 2, and $\psi_2 = s_2/\sqrt{2}$.

STEP 3 $s_{31} = \frac{1}{\sqrt{2}}(-1 + 1) = 0$ and $s_{32} = \frac{1}{\sqrt{2}}(-1 - 1) = -\sqrt{2}$. Then $g_3 = s_3 + \sqrt{2}\,\psi_2 = s_3 + s_2$. It is 1 on $[2, 3)$ and zero elsewhere, with energy 1, so $\psi_3 = g_3$.

STEP 4 $s_4 = s_1 + s_2 + s_3$, so $g_4 = 0$ and s_4 adds no axis. Its coordinates are $s_{41} = \frac{1}{\sqrt{2}}(1 + 1) = \sqrt{2}$, $s_{42} = \frac{1}{\sqrt{2}}(1 - 1) = 0$ and $s_{43} = \int_2^3 1\,dt = 1$.

SOLUTION $\mathbf{s}_1 = (\sqrt{2}, 0, 0)$, $\mathbf{s}_2 = (0, \sqrt{2}, 0)$, $\mathbf{s}_3 = (0, -\sqrt{2}, 1)$ and $\mathbf{s}_4 = (\sqrt{2}, 0, 1)$. The set spans $N = 3$ dimensions.

CHECK Energies from the vectors are 2, 2, 3 and 3, equal to $\int s_i^2\,dt$.

> COMMON ERROR
>
> Use one axis for each independent signal, not one for each signal, so $N \le M$. Here $g_4 = 0$, so $N = 3$, not $M = 4$.

The figure draws the four vectors in three dimensions.

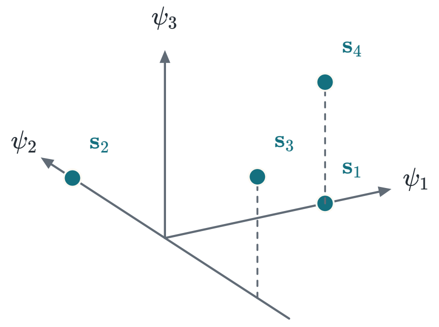


FIGURE 3.24 Four signals need three axes, seen here from an angle of 30°. $\mathbf{s}_4$ sits one unit above $\mathbf{s}_1$, along ψ_3.

The basis is not unique

Another orthonormal basis of the same space turns or reflects the axes. Taking the signals in another order in Gram–Schmidt gives such a basis.

The coordinates change. The quantities below do not.

> UNCHANGED
>
> $$N, \qquad \|\mathbf{s}_i\|^2 = E_i, \qquad \langle \mathbf{s}_i, \mathbf{s}_k \rangle, \qquad \|\mathbf{s}_i - \mathbf{s}_k\|$$
>
> Each is an integral of the waveforms, and the waveforms did not change.

The figure turns the axes by 30°. The coordinates of $\mathbf{s}_1 = (1, 1)$ become $(1.37, 0.37)$. The four points and their distances stay.

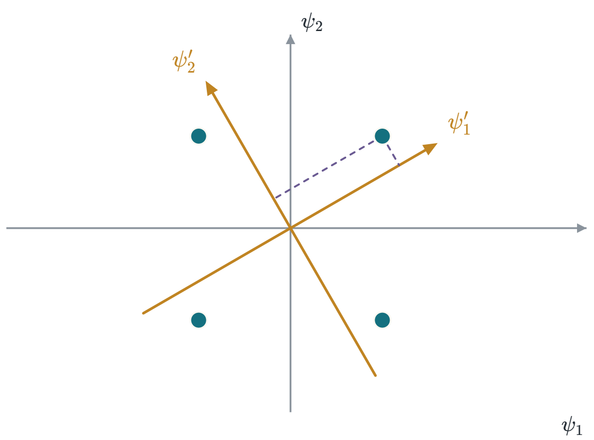


FIGURE 3.25 The axes ψ_1' and ψ_2' turned by 30°, where $\mathbf{s}_1 = (1.37, 0.37)$. The coordinates of $\mathbf{s}_1$ change. The four points and their distances do not.

For example, turn the axes by 45°. Then $\psi_1' = (\psi_1 + \psi_2)/\sqrt{2}$ and $\psi_2' = (\psi_2 - \psi_1)/\sqrt{2}$.

THE NEW COORDINATES OF $\mathbf{s}_1 = (1, 1)$

$$\langle \mathbf{s}_1, \psi_1' \rangle = \frac{1+1}{\sqrt{2}} = \sqrt{2}$$

$$\langle \mathbf{s}_1, \psi_2' \rangle = \frac{-1+1}{\sqrt{2}} = 0$$

ψ_1' points along $(1, 1)$, so all the length $\sqrt{2}$ of $\mathbf{s}_1$ lies on it. The new coordinates are $(\sqrt{2}, 0)$.

Gram–Schmidt around us

Four real systems use the Gram–Schmidt step.

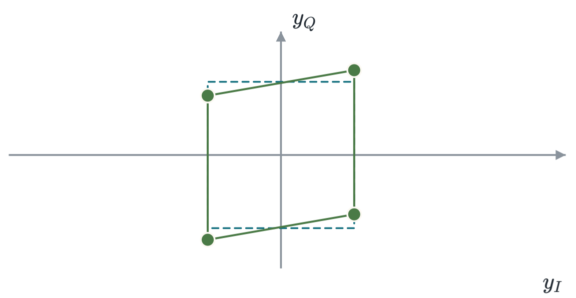


FIGURE 3.26 A receiver whose sine branch is $10°$ off reads QPSK as a sheared square, $y_Q = y\cos 10° + x\sin 10°$. Gram–Schmidt removes the $\sin 10° = 0.174$ leak and restores the dashed square.

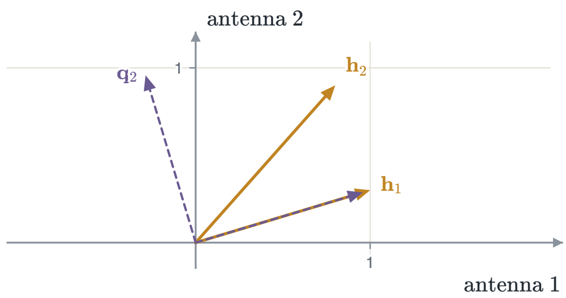


FIGURE 3.27 A two-antenna Wi-Fi receiver orthogonalizes the channel columns $\mathbf{h}_1 = (1, 0.3)$ and $\mathbf{h}_2 = (0.8, 0.9)$ before detection. The result $\mathbf{q}_1, \mathbf{q}_2$ is the QR factorization.

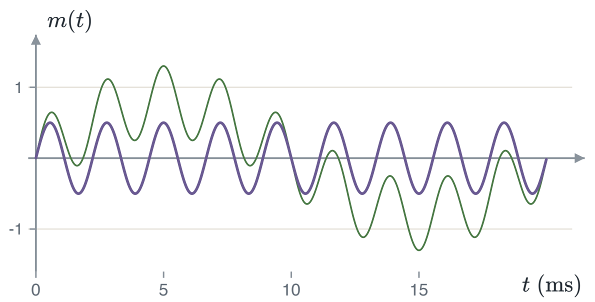


FIGURE 3.28 A hum filter projects the microphone signal $m(t)$ on a 50 Hz reference $r(t)$ over 20 ms. The coefficient $\langle m, r\rangle / \|r\|^2 = 0.8$, and $m - 0.8\, r$ leaves the 450 Hz tone.

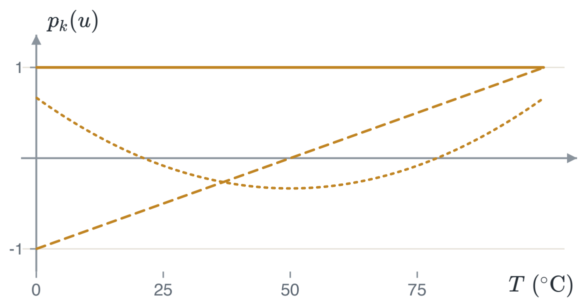


FIGURE 3.29 A sensor fit over 0 to 100 °C uses $u = (T - 50)/50$. Gram–Schmidt on 1, u, u^2 over $[-1, 1]$ gives the orthogonal set 1, u, $u^2 - \frac{1}{3}$.

Each case subtracts the projection on a known signal. That is step k of Gram–Schmidt, $g_k = s_k - \sum_i s_{ki}\psi_i$.

What is left is orthogonal to everything removed. It holds only what is new in the signal.

⚠ THE ORDER

Starting from $\mathbf{h}_2$ instead of $\mathbf{h}_1$ gives other axes. They span the same plane.

3.4 Summary

From waveforms to points

One waveform goes through the whole translation: basis, correlators and point. The figure uses $s(t) = 1.5\psi_1(t) - 0.8\psi_2(t)$ on the two unit pulses.

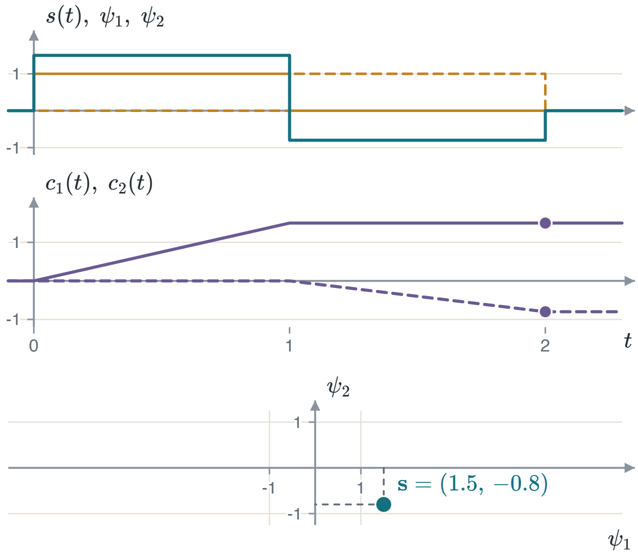


FIGURE 3.30 The waveform $s(t)$ over the basis ψ_1 and ψ_2 (dashed), the correlator outputs $c_1(t)$ and $c_2(t)$ (dashed), and the point. Each correlator integrates $s(t)\,\psi_j(t)$. Its value at $t = 2$ is one coordinate of the point.

The translation takes four moves.

1. Find a basis, by inspection or by Gram–Schmidt.
2. Correlate each waveform with each ψ_j.
3. Draw the vectors as points.
4. Read energies and distances off the picture.

The receiver of Chapter 4 correlates, as the analyzer does, and then picks a point. Its error probability depends only on the distances between the points.

For example, a set of $M = 16$ waveforms spans $N = 2$ dimensions. The receiver needs one correlator for each basis function, so $N = 2$ correlators.

Quick check

Six short predictions. Each answer follows with its reason.

1. Two pulses never overlap in time. What is their inner product?

Answer: 0. Where one is nonzero the other is zero, so the product is zero everywhere.

2. $\psi(t) = c$ on $[0, 4]$ has unit energy when c is what?

Answer: $1/2$. $4c^2 = 1$, so $c = 1/2$.

3. $s(t) = 2\psi_1(t) - 3\psi_2(t)$ on an orthonormal basis. What is its energy?

Answer: 13. $\|\mathbf{s}\|^2 = 2^2 + (-3)^2 = 13$.

4. $\mathbf{s}_1 = (1, 2)$ and $\mathbf{s}_2 = (4, 6)$. What is their distance?
Answer: 5. $\sqrt{3^2 + 4^2} = \sqrt{25} = 5$.

5. Five waveforms are all multiples of one pulse. What N does Gram–Schmidt give?
Answer: $N = 1$. After the first axis every remainder is zero.

6. QPSK has $E = 2$. What is its $d_{\min} = 2\sqrt{E}\sin(\pi/4)$?
Answer: 2. $2\sqrt{2}\,(0.707) = 2$.

Summary of results

TABLE 3.1 Summary of Chapter 3: the geometric representation of signal waveforms.

RESULT	STATEMENT	ANCHOR
Orthonormal set	functions with $\int \psi_j \psi_k \, dt = 1$ for $j = k$ and 0 for $j \neq k$: unit energy, and every pair orthogonal	PS CH8.1
Coordinate	$s_{ij} = \int_0^T s_i(t)\, \psi_j(t)\, dt$, one correlator for each basis function	PS CH8.1
Synthesis	$s_i(t) = \sum_{j=1}^{N} s_{ij}\, \psi_j(t)$. The vector $\mathbf{s}_i$ holds the whole waveform	PS CH8.1
Inner products	preserved: $\int x(t)\, y(t)\, dt = \sum_k x_k y_k$	PS CH8.1
Energy	$E_i = \lVert\mathbf{s}_i\rVert^2$, the squared distance from its point to the origin	PS CH8.1
Distance	$d_{ik} = \lVert\mathbf{s}_i - \mathbf{s}_k\rVert$, the root of $\int (s_i - s_k)^2\, dt$	PS CH8.1
Binary sets	antipodal is further apart, $d = 2\sqrt{E_b}$ against $\sqrt{2E_b}$. Orthogonal needs twice the energy, 3 dB, for the same distance	PS CH8.2
Carrier axes	a cosine and a sine are orthogonal over $[0, T]$ when $f_c T$ is a whole number. Scaled by $\sqrt{2/T}$ they are an orthonormal pair	PS CH8.6.1
M-PSK	$d_{\min} = 2\sqrt{E}\sin(\pi/M)$. All M points lie on the circle of radius $\sqrt{E}$	PS CH8.6.1
Gram–Schmidt step	$g_k = s_k - \sum_{i<k} s_{ki}\psi_i$, then $\psi_k = g_k/\sqrt{E_{g_k}}$. A zero g_k adds no axis	PS CH8.1
Dimension	$N \leq M$, with $N = M$ only when no signal is a combination of the others	PS CH8.1
Change of basis	only the coordinates change. N, the energies and the distances stay, and so do the receiver and its error probability	PS CH8.1

The method: find a basis, correlate to get the coordinates, and draw the constellation. Chapter 4 detects a received point by its distances to these points.

Projects to try

Four optional projects use the chapter on simulated signals. Each gives an aim, what it practises, steps and what to look for.

GRAM–SCHMIDT ON NOISY WAVEFORMS

Run Gram–Schmidt on sampled waveforms and see how noise changes the count of dimensions. It practises Gram–Schmidt on samples instead of formulas. It shows why a tolerance is needed to call a remainder zero, and how to count the dimensions of a signal set.

Steps: Sample the three pulses of Example 3.3 at 1000 points. Run Gram–Schmidt and print the energy of each remainder. Add a little Gaussian noise to each signal and run it again. Raise the tolerance until the count returns to 2.

Look for: without noise the third remainder is zero. With noise it is small but not zero, so a strict test finds three axes.

SHARE A CHANNEL WITH WALSH CODES

Send two users' bits at the same time and separate them with orthogonal codes. It practises Walsh codes as an orthogonal set, correlation as the analyzer, and what breaks when the codes lose their alignment.

Steps: Build the 8×8 Hadamard matrix and take two of its rows as codes. Map each user's bits to ± 1 and multiply by that user's code. Add the two signals and correlate the sum with each code. Delay one user by one chip and repeat.

Look for: aligned, each correlator returns only its own user's bit. With a one-chip delay the other user leaks in.

ORTHOGONAL CARRIERS WITH THE FFT

Place symbols on orthogonal subcarriers and recover them, as OFDM does. It practises sinusoids with whole cycles as an orthogonal basis, with the inverse FFT as the synthesizer and the FFT as the analyzer. It shows why a frequency offset breaks orthogonality.

Steps: Choose 16 QPSK symbols, one for each subcarrier. Build the time signal with an inverse FFT of length 64. Recover the symbols with an FFT and plot them. Shift the frequency by a tenth of the spacing and plot again.

Look for: the points come back exactly. With the offset each point spreads into a small cloud, because the carriers leak into each other.

A CONSTELLATION ZOO

Compare the smallest distance of many signal sets at the same average energy. It practises normalizing a constellation to average energy 1, computing $d_{\min}$ from all pairs of points, and the trade between bits a symbol and distance.

Steps: Build BPSK, QPSK, 8-PSK, 16-PSK, 16-QAM and 64-QAM. Scale each to average energy 1. Compute $d_{\min}$ for each and plot it against $\log_2 M$. Compare 16-PSK with 16-QAM.

Look for: $d_{\min}$ falls as bits are added. At 16 points, QAM keeps its points further apart than PSK.

CHAPTER

4

The optimal receiver in additive white Gaussian noise

One of M signals is sent and the channel adds noise. The receiver turns what arrives into a point and picks the nearest signal point.

A bank of correlators turns $r(t)$ into a point $\mathbf{r}$. The best rule picks the nearest signal point, with a handicap for each prior. How often the receiver is wrong depends only on the distances between the points. The union bound turns those distances into a number.

The figure below sends one point many times and marks the received points that land in the wrong region.

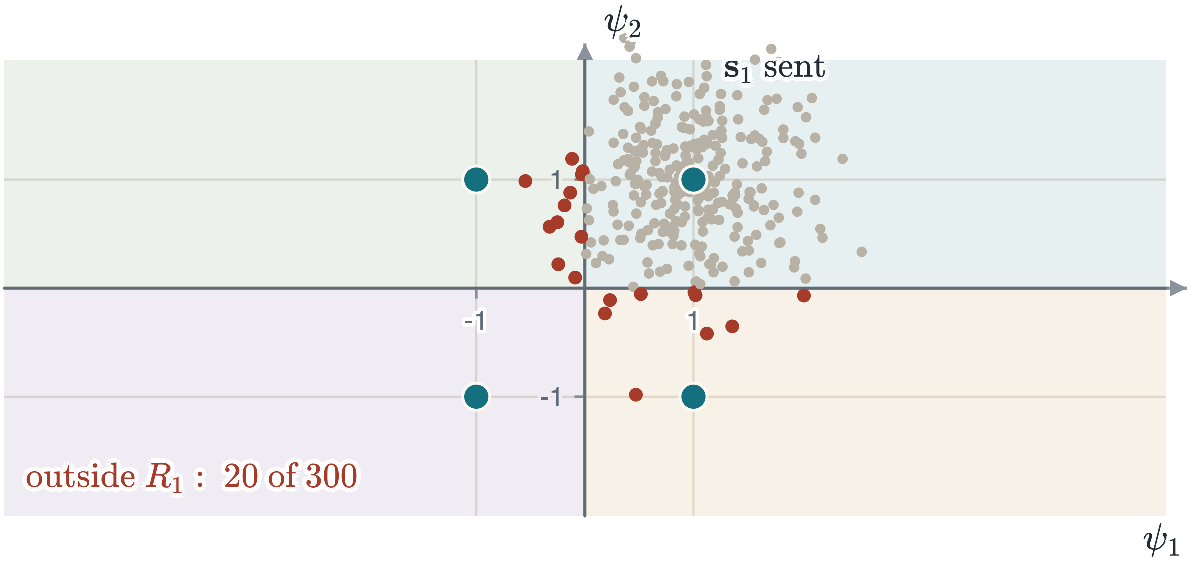


FIGURE 4.1 Four signal points and their decision regions. The grey dots are received points when $\mathbf{s}_1$ is sent, and the red ones fall outside its region R_1.

4.1 The observation

Bits into symbols

With M waveforms, each symbol carries $k = \log_2 M$ bits. The bit stream is read in blocks of k bits, and each block picks one waveform.

BITS AND SYMBOLS

$$k = \log_2 M, \qquad R_s = \frac{1}{T}, \qquad R_b = kR_s, \qquad T_b = \frac{T}{k}$$

Each symbol carries k bits and lasts T. The symbol rate R_s counts waveforms a second. The bit rate R_b counts bits.

Each T seconds the transmitter sends one of the M waveforms $s_1(t), \ldots, s_M(t)$. The receiver must name it from a noisy $r(t)$.

The figure below reads twelve bits in blocks of two. Each block picks one of four levels and holds it for $T = 2T_b$.

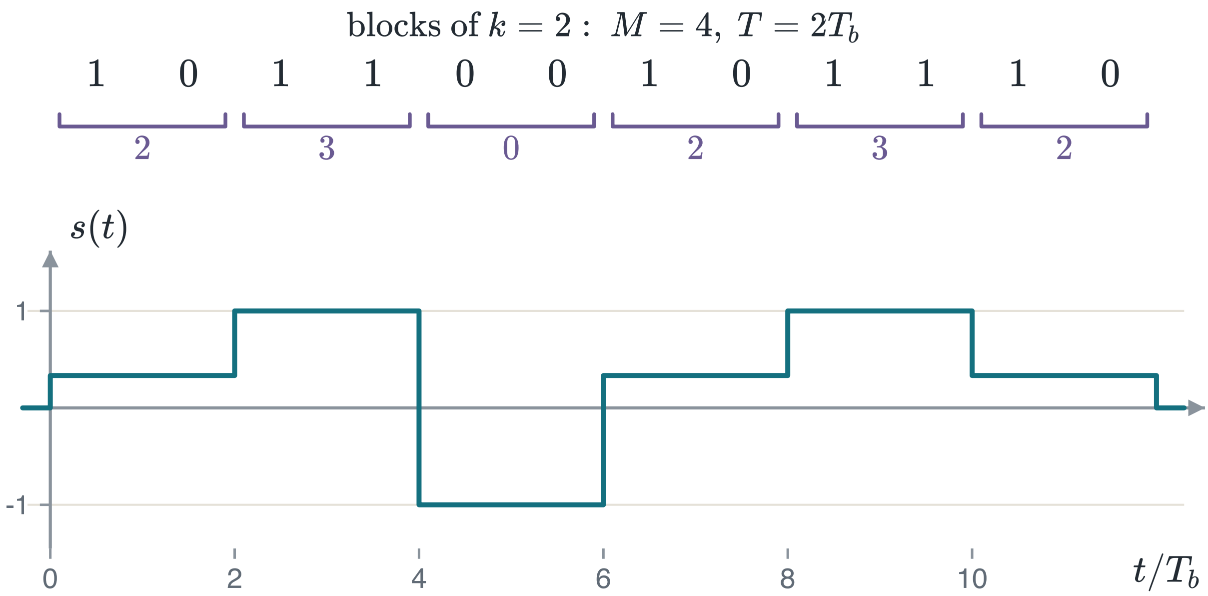


FIGURE 4.2 Twelve bits read in blocks of $k = 2$, so $M = 2^k = 4$ and $T = kT_b = 2T_b$. The number under each block is the level it picks, from 0 to 3.

For example, a modem sends 2400 symbols a second with $M = 16$. Then $k = \log_2 16 = 4$ bits a symbol, and $R_b = 4 \times 2400 = 9600$ b/s.

The observation vector

Over one symbol the receiver sees $r(t) = s_i(t) + n(t)$. Here $n(t)$ is white Gaussian noise of two-sided power spectral density $N_0/2$.

The receiver correlates $r(t)$ with each of the N orthonormal basis functions $\psi_1(t), \ldots, \psi_N(t)$ of the signal set. Each correlator multiplies by one basis function and integrates over one symbol.

CORRELATOR OUTPUTS

$$
\begin{aligned}
r_j &= \int_0^T r(t)\,\psi_j(t)\,dt \\
&= \int_0^T s_i(t)\,\psi_j(t)\,dt + \int_0^T n(t)\,\psi_j(t)\,dt \\
&= s_{ij} + n_j, \qquad j = 1, \ldots, N
\end{aligned}
$$

The integral is linear, so it splits over the sum. The first integral is the coordinate s_{ij} of the sent signal. The second is a noise coordinate n_j.

Stack the N outputs into one vector. The result is the observation the receiver decides from.

THE OBSERVATION VECTOR

$$\mathbf{r} = \mathbf{s}_i + \mathbf{n}$$

The receiver now works with one point $\mathbf{r}$. The noise moves it away from the sent point $\mathbf{s}_i$.

The figure below follows one received waveform, with $T = 1$, through two correlators.

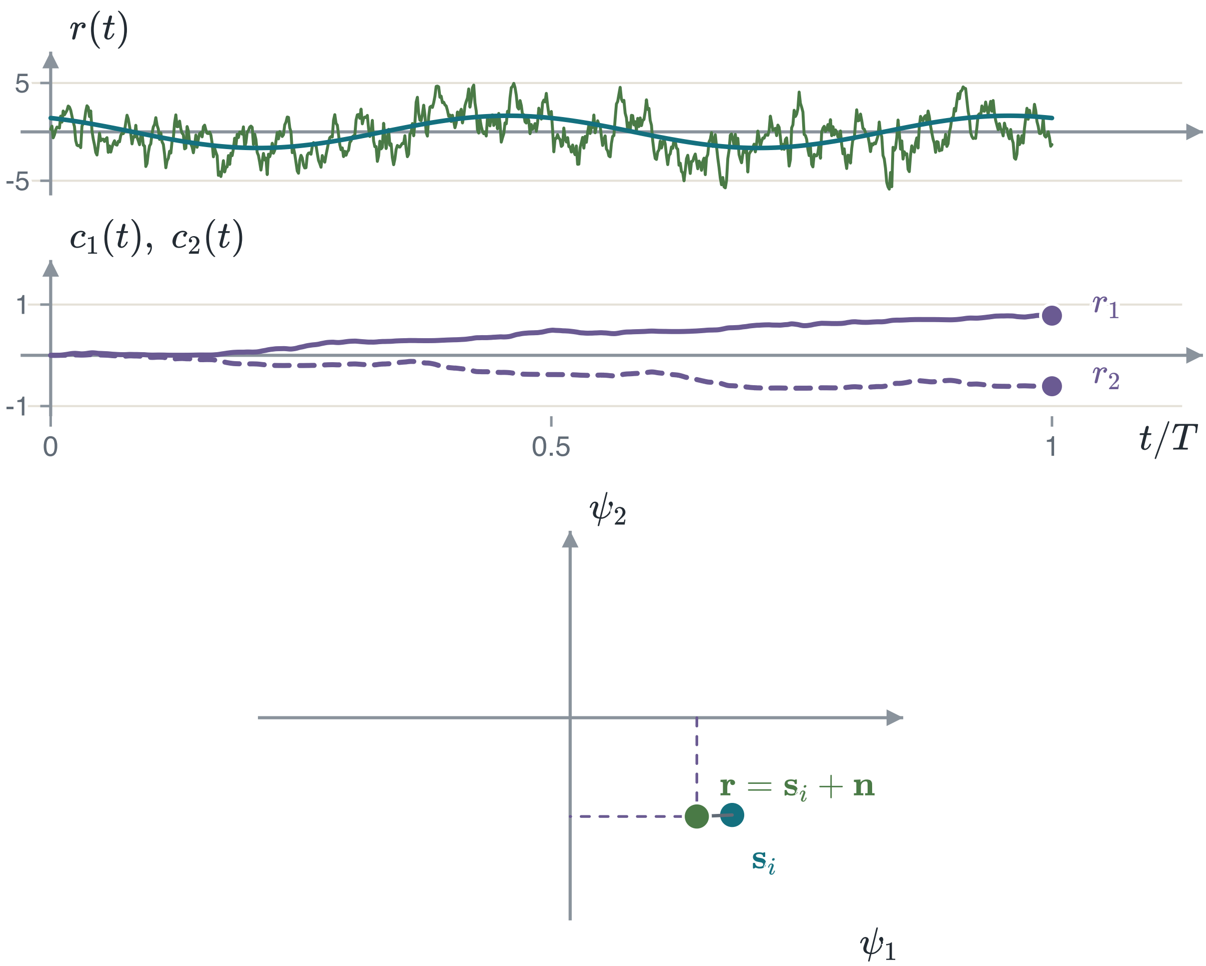


FIGURE 4.3 Top: $r(t)$, green, is $s_i(t)$, cyan, plus noise. Middle: the running integrals $c_1(t)$, solid, and $c_2(t)$, dashed, of $r(t)\,\psi_j(t)$. Bottom: their end values r_1 and r_2 are the coordinates of $\mathbf{r}$.

For example, let $\mathbf{s}_i = (2, -1)$ be sent with noise coordinates $n_1 = 0.3$ and $n_2 = -0.4$. Then $\mathbf{r} = \mathbf{s}_i + \mathbf{n} = (2 + 0.3,\ -1 - 0.4) = (2.3, -1.4)$.

The matched-filter bank

A bank of matched filters gives the same numbers as the correlators. The filter for axis j is its basis function turned around in time.

MATCHED FILTER

$$h_j(t) = \psi_j(T - t)$$

The filter output is the convolution of $r(t)$ with $h_j(t)$. Write the convolution, then put in $h_j(t - \tau) = \psi_j(T - t + \tau)$.

$$\begin{aligned} y_j(t) &= \int_0^T r(\tau)\, h_j(t - \tau)\, d\tau \\ &= \int_0^T r(\tau)\, \psi_j(T - t + \tau)\, d\tau \end{aligned}$$

Now sample the output at $t = T$. The shift $T - t$ becomes zero, and the integral is the correlator.

SAMPLE AT T

$$y_j(T) = \int_0^T r(\tau)\, \psi_j(\tau)\, d\tau = r_j$$

At any other time the two outputs differ.

The figure below uses $\psi(t) = \sqrt{3}\,t$ on $[0, 1]$ and $r(t) = 1.5\,\psi(t)$, with no noise. The two curves meet only at $t = T = 1$.

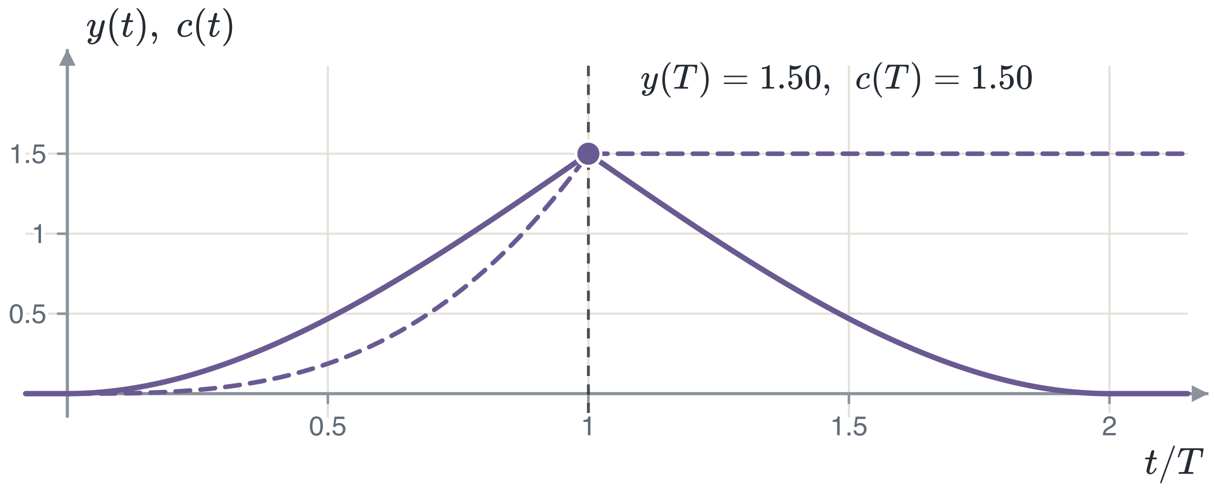


FIGURE 4.4 Solid: the matched filter output $y(t)$. Dashed: the correlator output $c(t)$. At the sampling time $t_0 = T$ both equal 1.5.

The value at T follows from the energy of ψ. The correlator gives $y(1) = 1.5 \int_0^1 \psi^2(t)\, dt = 1.5 \int_0^1 3t^2\, dt = 1.5$.

The noise the receiver ignores

Noise has parts in every direction, not only in the signal space. The part outside the span of $\psi_1, \ldots, \psi_N$ is written $n'(t)$. It does not depend on the signal sent.

Let $\tilde{\mathbf{r}}$ be the received waveform with nothing dropped. It is $\mathbf{r}$ plus n', and n' is at right angles to the signal space. So Pythagoras splits each squared distance into two parts.

$$\tilde{\mathbf{r}} - \mathbf{s}_j = (\mathbf{r} - \mathbf{s}_j) + n'$$

THE NOISE OUTSIDE THE SIGNAL SPACE

$$\|\tilde{\mathbf{r}} - \mathbf{s}_j\|^2 = \|\mathbf{r} - \mathbf{s}_j\|^2 + \|n'\|^2$$

The second part is the same for every j. It cannot change which signal is nearest, so the receiver can drop it.

SUFFICIENT STATISTICS

The N correlator outputs $r_1, \ldots, r_N$ hold everything the decision needs. Nothing is lost by dropping $n'(t)$. Outputs with this property are called **sufficient statistics**.

The figure below draws the signal space as a sheet in three dimensions.

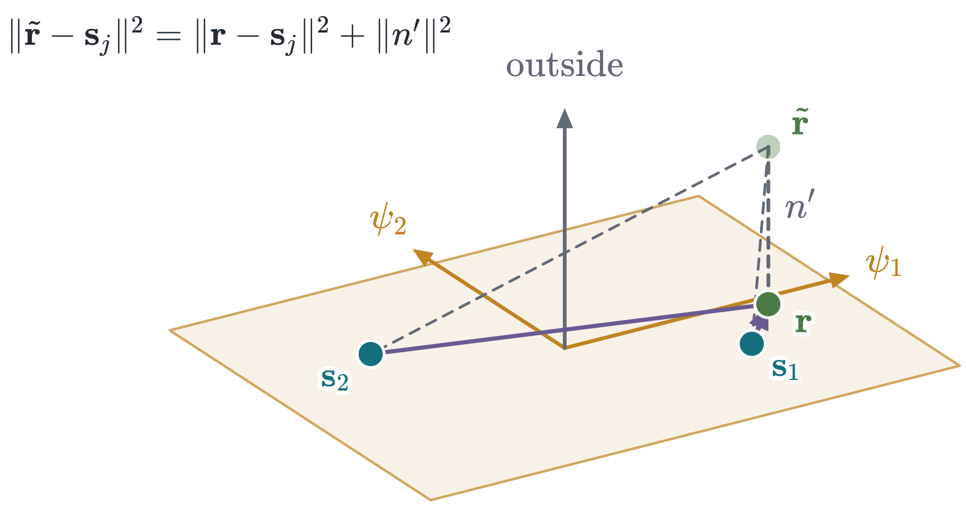


FIGURE 4.5 The amber sheet is the signal space, spanned by ψ_1 and ψ_2. The noise splits into a part in the sheet and n' outside it. Only the part in the sheet moves $\mathbf{r}$.

For example, let the squared distances in the sheet be $\|\mathbf{r} - \mathbf{s}_1\|^2 = 0.5$ and $\|\mathbf{r} - \mathbf{s}_2\|^2 = 2.0$, with $\|n'\|^2 = 0.8$. The full squared distances are $0.5 + 0.8 = 1.3$ and $2.0 + 0.8 = 2.8$. The gap stays 1.5, so $\mathbf{s}_1$ still wins.

The noise components

Each noise coordinate is the noise correlated with one basis function. It is a weighted integral of a Gaussian process, so it is Gaussian.

ONE COMPONENT

$$n_j = \int_0^T n(t)\,\psi_j(t)\,dt, \qquad E[n_j] = 0$$

The noise has mean zero, and the integral keeps the mean at zero.

The variance of one component, and the correlation of two, follow from the autocorrelation of white noise, $E[n(t)n(u)] = \frac{N_0}{2}\delta(t-u)$.

$$\begin{aligned} E[n_j n_k] &= \int_0^T\int_0^T E[n(t)n(u)]\,\psi_j(t)\,\psi_k(u)\,dt\,du \\ &= \int_0^T\int_0^T \frac{N_0}{2}\,\delta(t-u)\,\psi_j(t)\,\psi_k(u)\,dt\,du \\ &= \frac{N_0}{2}\int_0^T \psi_j(t)\,\psi_k(t)\,dt \end{aligned}$$

The delta function collapses the integral over u. Orthonormality then gives 1 for the last integral when $j = k$, and 0 otherwise.

TWO COMPONENTS

$$E[n_j^2] = \frac{N_0}{2}, \qquad E[n_j n_k] = 0, \quad j \neq k$$

Gaussian variables that are uncorrelated are independent. So the N components are independent, each with variance $N_0/2$.

Every axis has the same variance, and the axes are independent. So the cloud of received points around a signal point is round.

⚠ AN ELLIPSE

Coloured noise gives correlated components, with correlation coefficient $\rho \neq 0$. The cloud tilts into an ellipse, and the nearest point is no longer the best guess.

For example, noise with $\rho = 0.8$ reaches the two correlators. A positive ρ makes n_2 follow n_1, so the cloud stretches along the line $n_1 = n_2$.

The figure below draws two components of variance 1 for two values of ρ.

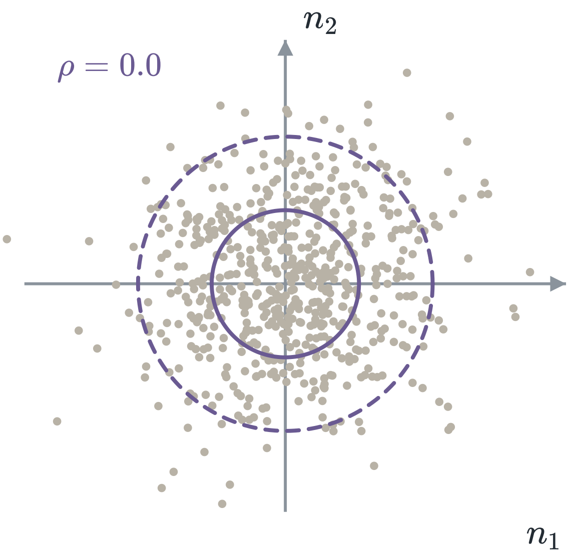


FIGURE 4.6 $\rho = 0$: independent components and a round cloud. The solid and dashed curves hold about 39% and 86% of the points.

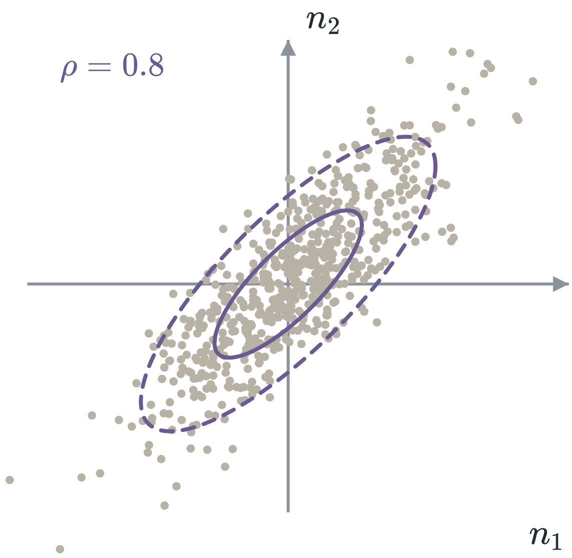


FIGURE 4.7 $\rho = 0.8$: n_2 follows n_1, and the cloud stretches along the line $n_1 = n_2$.

EXAMPLE 4.1 — FOUR-LEVEL PAM

GIVEN Four-level PAM, $s_m \in \{-1.5, -0.5, 0.5, 1.5\}$ on a unit-energy pulse, with $N_0 = 0.1$.

FIND The conditional density $f(r \mid s_m)$ of the correlator output for each level, and the height of each bell.

METHOD One basis function means one correlator and one number, $r = s_m + n$. The noise n is Gaussian with mean 0 and variance $\sigma^2 = N_0/2$. So r is Gaussian and centred on s_m.

SOLUTION Put $\sigma^2 = N_0/2$ into the Gaussian density.

$$f(r \mid s_m) = \frac{1}{\sqrt{2\pi\sigma^2}}\, e^{-(r-s_m)^2/2\sigma^2} = \frac{1}{\sqrt{\pi N_0}}\, e^{-(r-s_m)^2/N_0}$$

Here $\sigma^2 = 0.05$. The peak is $1/\sqrt{\pi N_0} = 1/\sqrt{0.1\pi} = 1.78$. The four bells have one shape, each centred on its level.

CHECK Each bell has $\sigma = \sqrt{0.05} = 0.22$, well inside half the spacing between levels, 0.5. At $N_0 = 0.6$, $\sigma = 0.55$ and neighbouring bells overlap.

COMMON ERROR

Use $N_0/2$, not N_0, as the variance. N_0 makes each bell $\sqrt{2}$ times too wide.

The figure below draws the four densities of Example 4.1.

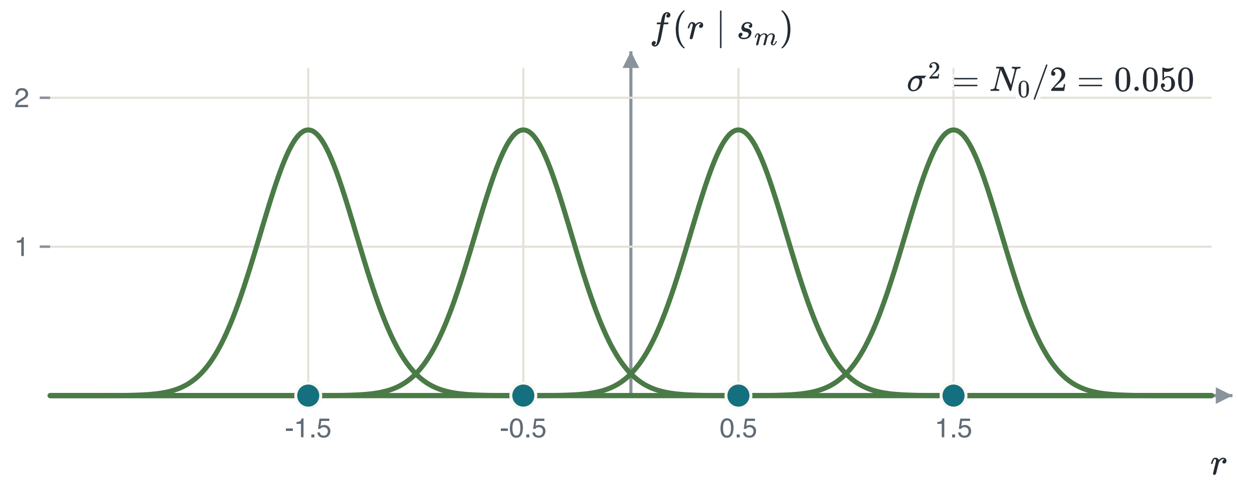


FIGURE 4.8 The four conditional densities of Example 4.1, drawn for $N_0 = 0.1$. Each bell sits on one level, and its width grows with $\sqrt{N_0}$.

EXAMPLE 4.2 — FOUR ORTHOGONAL SIGNALS

GIVEN Four orthogonal signals of energy $E = 4$, with $N_0 = 1$. $\mathbf{s}_1 = (2, 0, 0, 0)$ is sent.

FIND The density of each correlator output r_k, and the mean of r_1.

METHOD Each signal lies along its own axis, at distance $\sqrt{E}$ from the origin. So only the first correlator sees the signal.

$$r_1 = \sqrt{E} + n_1, \qquad r_k = n_k, \quad k = 2, 3, 4$$

SOLUTION Each n_k is Gaussian with variance $N_0/2 = 0.5$, so $2\sigma^2 = 1$.

$$f(r_1 \mid \mathbf{s}_1) = \frac{e^{-(r_1-2)^2}}{\sqrt{\pi}}, \qquad f(r_k \mid \mathbf{s}_1) = \frac{e^{-r_k^2}}{\sqrt{\pi}}$$

The mean of r_1 is $\sqrt{E} = 2$. Each density has peak $1/\sqrt{\pi} = 0.56$, and the four outputs are independent.

CHECK The means $(2, 0, 0, 0)$ have squared length $2^2 = 4 = E$, as they must.

COMMON ERROR

Use $\sqrt{E} = 2$, not $E = 4$, as the mean of r_1. The coordinate of $\mathbf{s}_1$ is its length, and E is its squared length.

The figure below draws the four densities of Example 4.2.

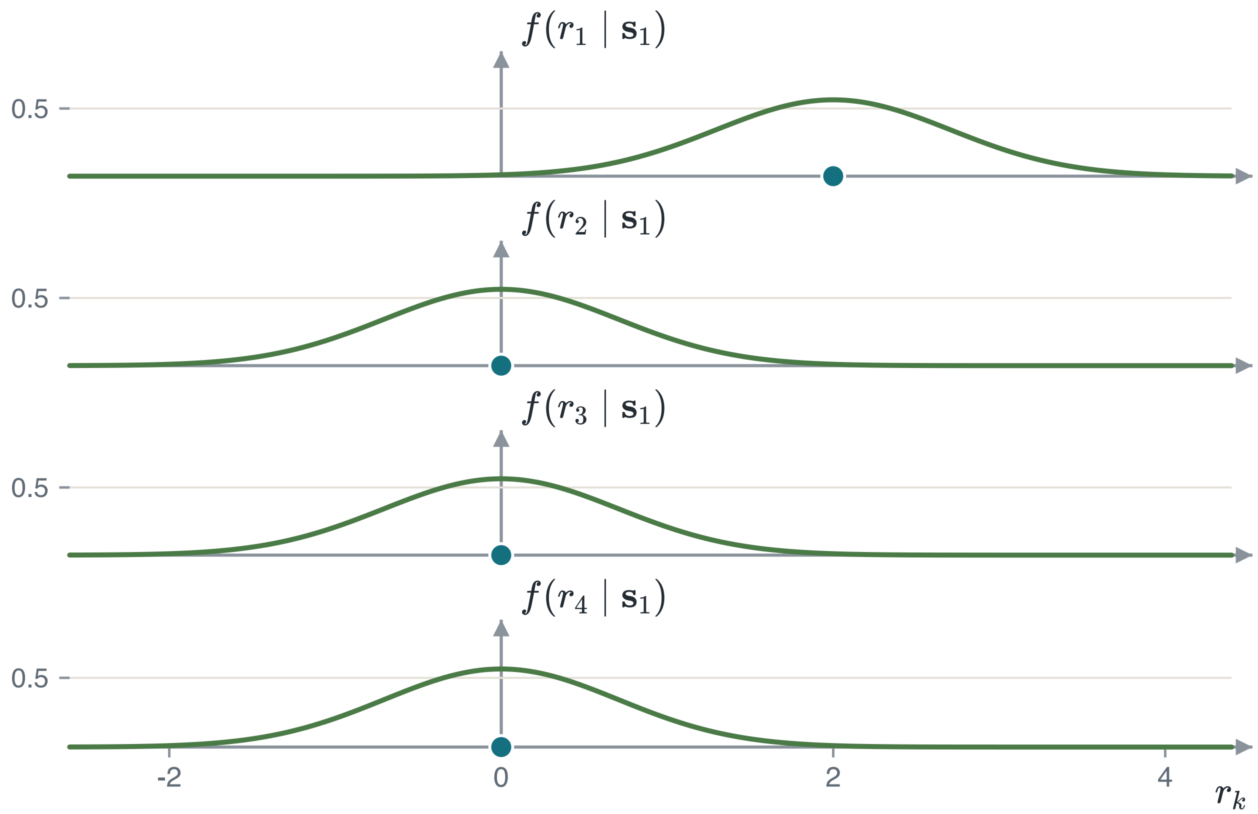


FIGURE 4.9 The four correlator outputs of Example 4.2 when $\mathbf{s}_1$ is sent. Only r_1 is centred away from zero.

Correlators around us

Four everyday receivers correlate what arrives with a known waveform.

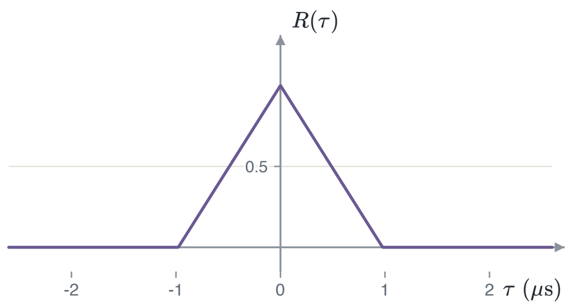


FIGURE 4.10 A GPS receiver correlates the C/A code with a local copy. $R(\tau) = 1 - |\tau|/T_c$, with $T_c = 0.978\ \mu$s, peaks when the two align.

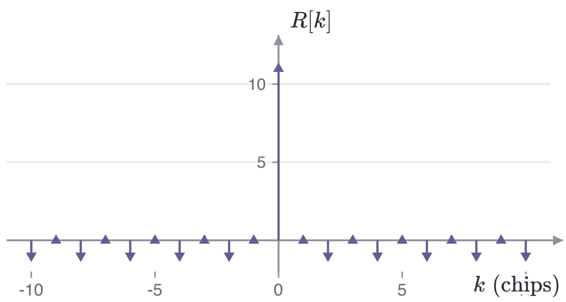


FIGURE 4.11 802.11b Wi-Fi spreads each bit with the 11-chip Barker code. Its correlation $R[k]$ is 11 at $k = 0$ and 0 or -1 elsewhere.

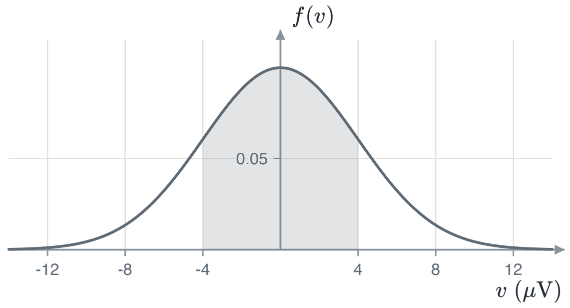


FIGURE 4.12 A $50\ \Omega$ resistor at 290 K, over 20 MHz, gives noise with $\sigma = \sqrt{4kTRB} = 4.0\ \mu$V. Every correlator output carries some.

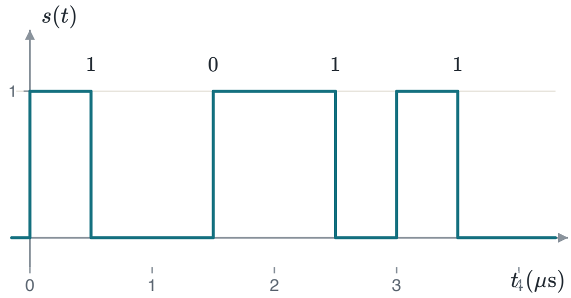


FIGURE 4.13 Aircraft ADS-B sends a 1 as a $0.5\ \mu$s pulse in the first half of the bit, a 0 in the second. Two correlators give (r_1, r_2).

> ⓘ CORRELATE, THEN DECIDE
>
> Each receiver multiplies what arrives by a known waveform and integrates. The number it gets is one coordinate of $\mathbf{r}$.

> ⓘ NOISE IN EVERY OUTPUT
>
> Thermal noise reaches every correlator. Each output is the signal coordinate plus a Gaussian sample.

> ⚠ THE WINDOW
>
> The integral runs over one symbol. A GPS copy that is late by T_c gives $R = 0$ and loses the whole signal.

4.2 The decision rule

MAP and ML

The receiver should name the signal that is most probable once $\mathbf{r}$ has been seen. This is the maximum a posteriori, or **MAP**, rule.

Bayes' rule writes that probability with the prior $P(\mathbf{s}_i)$ and the likelihood $f(\mathbf{r} \mid \mathbf{s}_i)$.

$$P(\mathbf{s}_i \mid \mathbf{r}) = \frac{P(\mathbf{s}_i)\, f(\mathbf{r} \mid \mathbf{s}_i)}{f(\mathbf{r})}$$

The denominator $f(\mathbf{r})$ is the same for every i, so it cannot change which i wins.

> MAP
>
> $$\hat{s} = \arg\max_i P(\mathbf{s}_i \mid \mathbf{r}) = \arg\max_i P(\mathbf{s}_i)\, f(\mathbf{r} \mid \mathbf{s}_i)$$

With equal priors the factor $P(\mathbf{s}_i)$ is common to every i and drops too. MAP then becomes the maximum-likelihood, or **ML**, rule.

ML

$$\hat{s} = \arg\max_i f(\mathbf{r} \mid \mathbf{s}_i)$$

For example, let $P(s_1) = 0.8$ and $P(s_2) = 0.2$. At some r the likelihoods are $f(r \mid s_1) = 0.2$ and $f(r \mid s_2) = 0.5$. MAP compares $0.8(0.2) = 0.16$ with $0.2(0.5) = 0.10$ and picks s_1. ML picks s_2, because $0.5 > 0.2$.

In one dimension the MAP rule picks the taller of the two weighted densities $P(s_i)\, f(r \mid s_i)$. The figure below shows that their crossing τ is the threshold.

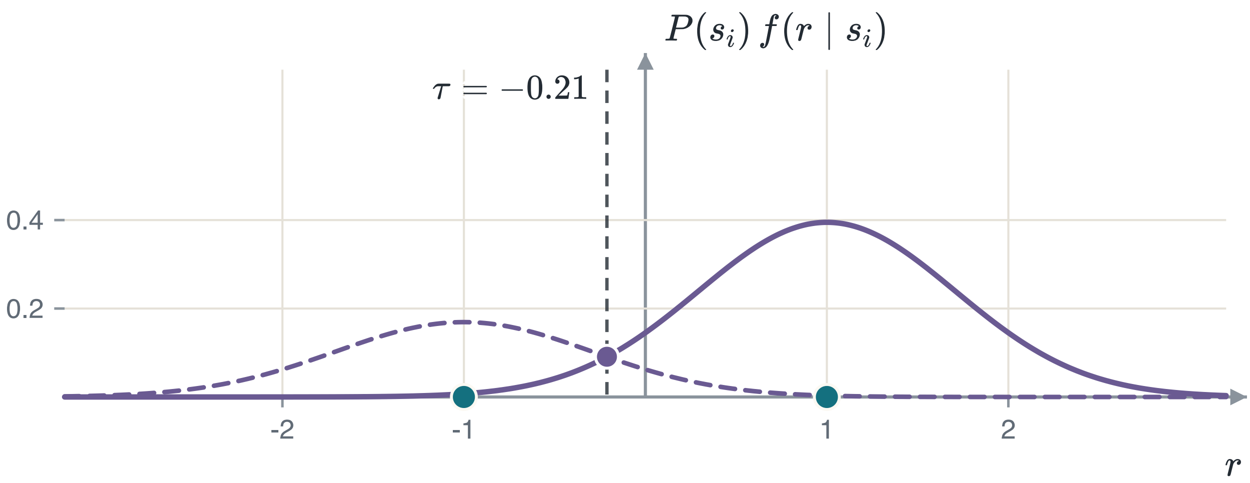


FIGURE 4.14 $s_1 = +1$ and $s_2 = -1$, with $N_0 = 1$, drawn for $P(s_1) = 0.7$. Solid: $P(s_1)\, f(r \mid s_1)$. Dashed: $P(s_2)\, f(r \mid s_2)$. The crossing τ sits on the side of the less likely s_2.

Minimum distance

In Gaussian noise the likelihood has a simple form. The N noise components are independent, so the density of $\mathbf{r}$ is a product of N bells.

$$\begin{aligned} f(\mathbf{r} \mid \mathbf{s}_i) &= \prod_{j=1}^{N} \frac{1}{\sqrt{\pi N_0}}\, e^{-(r_j - s_{ij})^2/N_0} \\ &= \frac{1}{(\pi N_0)^{N/2}}\, e^{-\|\mathbf{r}-\mathbf{s}_i\|^2/N_0} \end{aligned}$$

The sum of the squares in the exponent is the squared distance $\|\mathbf{r} - \mathbf{s}_i\|^2$. Now take the logarithm of the MAP product. The logarithm is increasing, so the best signal does not change.

$$\ln\big(P(\mathbf{s}_i)\, f(\mathbf{r} \mid \mathbf{s}_i)\big) = \ln P(\mathbf{s}_i) - \frac{N}{2}\ln(\pi N_0) - \frac{\|\mathbf{r} - \mathbf{s}_i\|^2}{N_0}$$

The middle term is the same for every i and drops. Multiply what is left by $-N_0$, which turns the largest into the smallest.

MINIMUM DISTANCE

$$\hat{s} = \arg\min_i \Big(\|\mathbf{r} - \mathbf{s}_i\|^2 - N_0 \ln P(\mathbf{s}_i) \Big)$$

With equal priors the MAP receiver picks the signal point nearest to $\mathbf{r}$. Unequal priors subtract $N_0 \ln P(\mathbf{s}_i)$ from each squared distance, a handicap in favour of the likely signals.

The figure below takes $\mathbf{r} = (0.5, 0.3)$ and four equally likely points $(\pm 1, \pm 1)$. It marks the squared distance D_i^2 to each point.

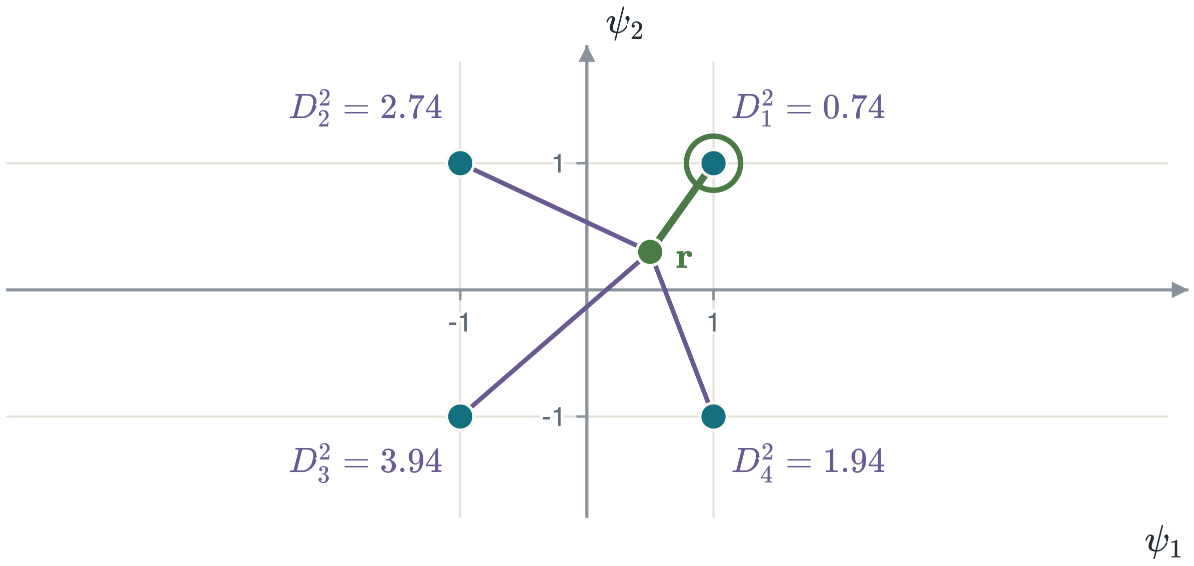


FIGURE 4.15 The received point $\mathbf{r} = (0.5, 0.3)$ and the points $\mathbf{s}_1 = (1, 1)$, $\mathbf{s}_2 = (-1, 1)$, $\mathbf{s}_3 = (-1, -1)$ and $\mathbf{s}_4 = (1, -1)$. The ring marks the nearest one, $\mathbf{s}_1$.

For $\mathbf{s}_1$ the squared distance is $D_1^2 = (0.5 - 1)^2 + (0.3 - 1)^2 = 0.25 + 0.49 = 0.74$. The others are 2.74, 3.94 and 1.94. With equal priors the receiver picks $\mathbf{s}_1$.

Now let $N_0 = 1$, with $P(\mathbf{s}_4) = 0.7$ and 0.1 for each of the others. Each score adds $-\ln P(\mathbf{s}_i)$ to the squared distance.

$$\begin{aligned}
\mathbf{s}_1 &: \quad 0.74 - \ln 0.1 = 0.74 + 2.30 = 3.04 \\
\mathbf{s}_2 &: \quad 2.74 + 2.30 = 5.04 \\
\mathbf{s}_3 &: \quad 3.94 + 2.30 = 6.24 \\
\mathbf{s}_4 &: \quad 1.94 - \ln 0.7 = 1.94 + 0.36 = 2.30
\end{aligned}$$

The smallest score is 2.30, so MAP picks $\mathbf{s}_4$.

The correlation metric

Expanding the squared distance gives the form in which a receiver is built.

$$\begin{aligned}\|\mathbf{r}-\mathbf{s}_i\|^2 &= \sum_{j=1}^{N}(r_j - s_{ij})^2 \\ &= \sum_{j=1}^{N} r_j^2 - 2\sum_{j=1}^{N} r_j s_{ij} + \sum_{j=1}^{N} s_{ij}^2 \\ &= \|\mathbf{r}\|^2 - 2\,\mathbf{r}\cdot\mathbf{s}_i + E_i\end{aligned}$$

Here $E_i = \|\mathbf{s}_i\|^2$ is the energy of $s_i(t)$. The term $\|\mathbf{r}\|^2$ is the same for every i and drops. Halve the rest and change its sign, so the smallest becomes the largest.

CORRELATION METRIC

$$\hat{s} = \arg\max_i \left(\mathbf{r}\cdot\mathbf{s}_i - \frac{E_i}{2} + \frac{N_0}{2}\ln P(\mathbf{s}_i)\right)$$

This is the same rule as minimum distance. With equal energies and equal priors only $\mathbf{r}\cdot\mathbf{s}_i$ is left: pick the largest correlation.

With unequal energies the term $-E_i/2$ must stay. Take 4-PAM at $s_i = -3, -1, 1, 3$ with $r = 1.6$ and equal priors. Each metric is $rs_i - s_i^2/2$.

$$\begin{aligned} s=-3:&\quad -4.8-4.5=-9.3\\ s=-1:&\quad -1.6-0.5=-2.1\\ s=1:&\quad 1.6-0.5=1.1\\ s=3:&\quad 4.8-4.5=0.3\end{aligned}$$

The largest is 1.1, so the receiver picks $s=1$, the nearest level. The correlation alone is largest for $s=3$, at 4.8. The figure below draws both sets of bars.

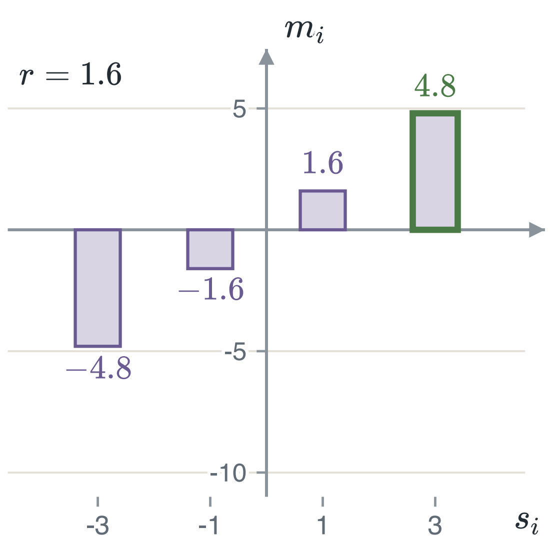


FIGURE 4.16 The correlation rs_i alone. The largest bar, outlined in green, is $s=3$.

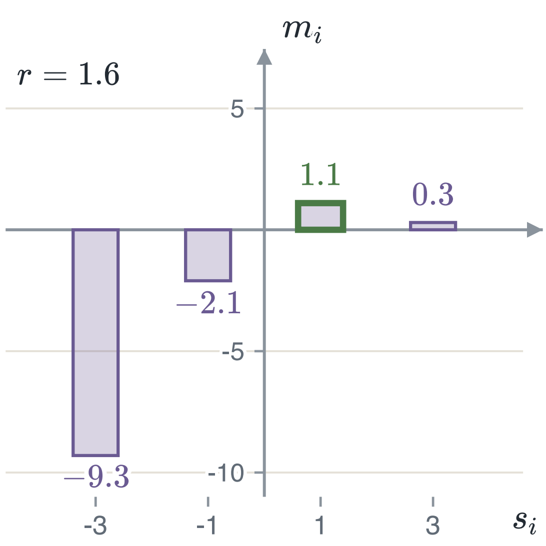


FIGURE 4.17 The metric $rs_i - s_i^2/2$. The energy term gives the win to $s=1$.

Two receiver structures

There are two ways to build the receiver. Both compute the correlation metric for every signal and keep the largest.

Method I computes $\mathbf{r} = (r_1, \ldots, r_N)$ with N correlators. Then it forms $\mathbf{r} \cdot \mathbf{s}_i + a_i$ for each i and keeps the largest.

> THE BIAS
>
> $$a_i = \frac{N_0}{2} \ln P(\mathbf{s}_i) - \frac{E_i}{2}$$
>
> There is one constant for each signal. The priors and the energies fix it before any signal arrives.

Method II correlates $r(t)$ with each waveform $s_i(t)$ directly. Inner products are preserved, so each of these correlators gives $\mathbf{r} \cdot \mathbf{s}_i$ at once.

$$\int_0^T r(t)\, s_i(t)\, dt = \mathbf{r} \cdot \mathbf{s}_i$$

Method I needs N correlators and Method II needs M. So Method I is cheaper whenever $N < M$. For example, $M = 16$ signals in $N = 2$ dimensions need 16 correlators in Method II and 2 in Method I.

The figure below draws both structures as block diagrams.

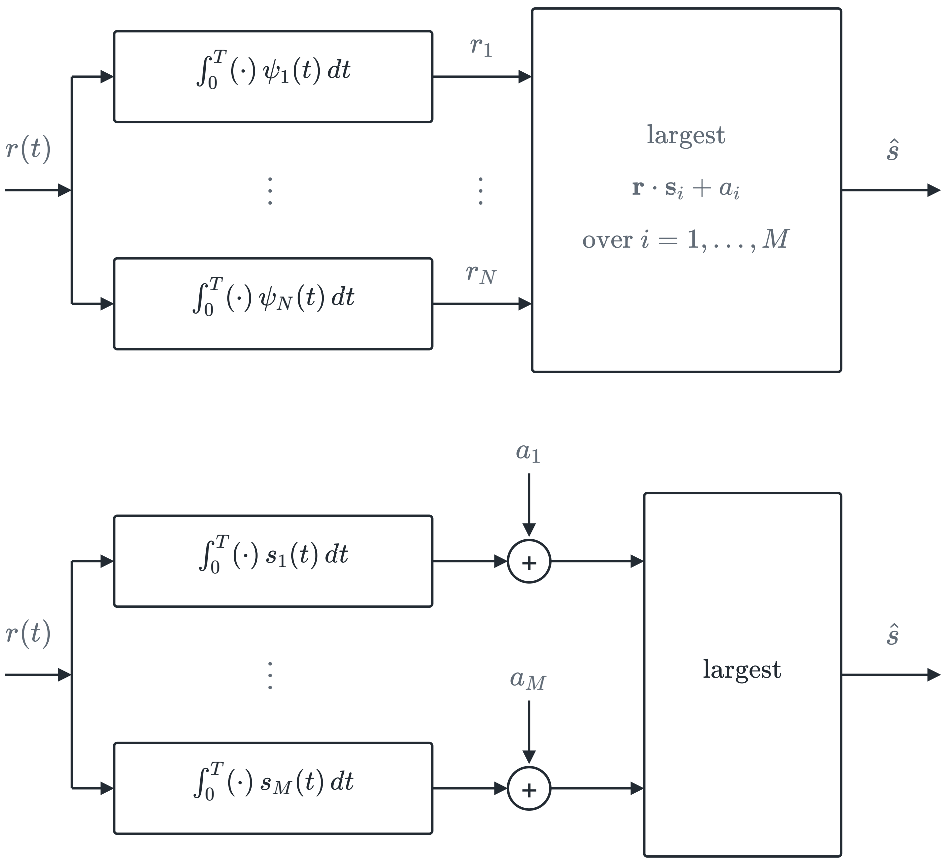


FIGURE 4.18 Top: Method I correlates with the N basis functions, then forms M metrics. Bottom: Method II correlates with the M waveforms and adds each bias a_i. Both give the same $\hat{s}$.

Optimality of the MAP rule

No other rule has a smaller error probability than MAP. In one dimension the reason is a picture, because the error probability is an area.

Take two signals and a threshold τ, with s_1 decided above τ. Each way to be wrong is a weighted density on the wrong side of τ.

ERROR AS AN AREA

$$P_e(\tau) = P(s_1)\int_{-\infty}^{\tau} f(r \mid s_1)\,dr + P(s_2)\int_{\tau}^{\infty} f(r \mid s_2)\,dr$$

Move τ right by a small Δ. The first area gains $P(s_1)f(\tau \mid s_1)\Delta$, and the second loses $P(s_2)f(\tau \mid s_2)\Delta$. So the slope of P_e is the difference.

$$\frac{dP_e}{d\tau} = P(s_1)\,f(\tau \mid s_1) - P(s_2)\,f(\tau \mid s_2)$$

The slope is zero where the two weighted densities are equal. That point is the MAP threshold, and no move of τ helps there.

IN ANY DIMENSION

Give each $\mathbf{r}$ to the signal whose weighted density $P(\mathbf{s}_i)f(\mathbf{r} \mid \mathbf{s}_i)$ is largest there. Any other choice adds area, so MAP has the smallest P_e.

The figure below uses ± 1 with $P(s_1) = 0.25$ at $+1$ and $N_0 = 0.5$. Its lower panel is $P_e(\tau)$ as the threshold moves.

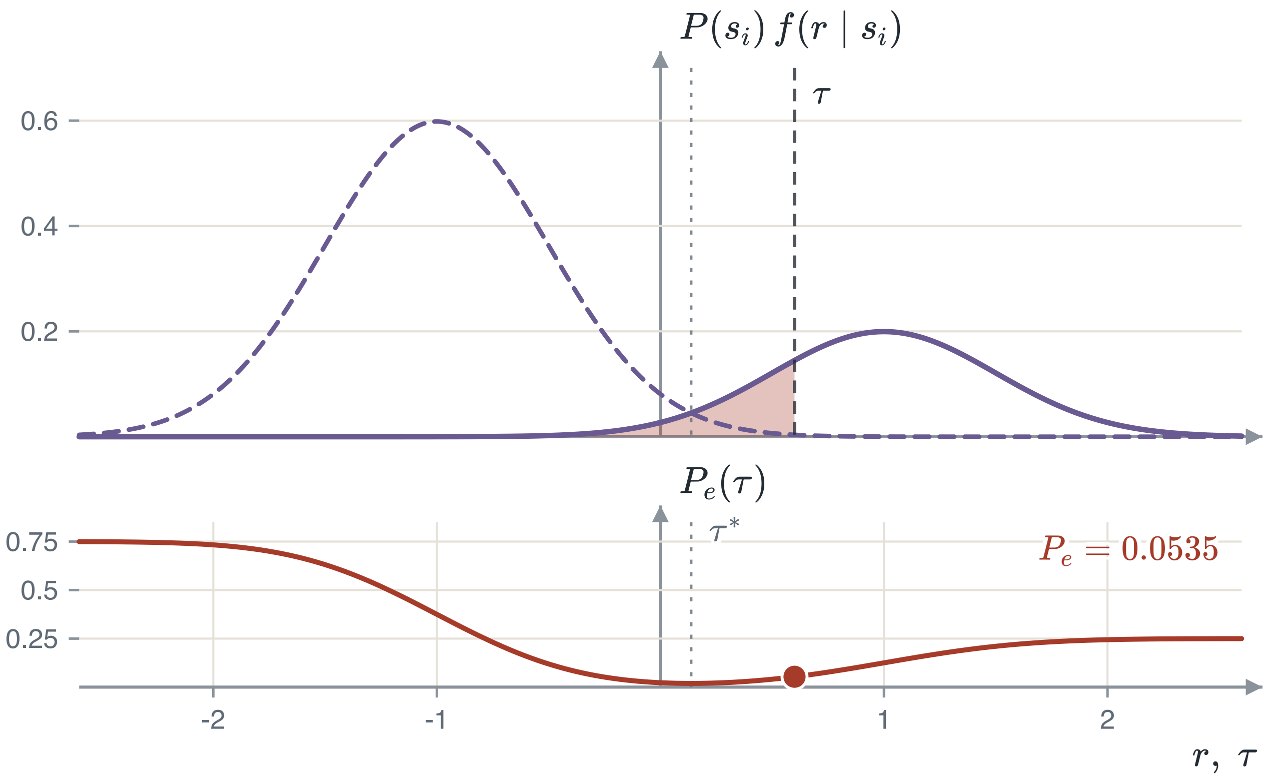


FIGURE 4.19 Drawn for $\tau = 0.6$. Top: the red areas are the weighted densities on the wrong side of τ. Bottom: their sum $P_e(\tau)$, smallest at $\tau^* = 0.137$, where the weighted densities cross.

The minimum sits at $\tau^* = 0.137$, with $P_e = 0.0192$. At $\tau = 0$ it is 0.0228.

EXAMPLE 4.3 — A MAP THRESHOLD

GIVEN Antipodal signals ± 1, so $E_b = 1$, with $P(s_1) = 0.25$ at $+1$, $P(s_2) = 0.75$ at -1, and $N_0 = 0.5$.

FIND The MAP threshold τ, the direction it moves, and the error probability.

METHOD The MAP threshold is where the two weighted densities are equal. Set them equal at $r = \tau$ and take logarithms. The common factor $1/\sqrt{\pi N_0}$ cancels.

$$\begin{aligned}
\ln P(s_1) - \frac{(\tau - \sqrt{E_b})^2}{N_0} &= \ln P(s_2) - \frac{(\tau + \sqrt{E_b})^2}{N_0} \\
\frac{(\tau + \sqrt{E_b})^2 - (\tau - \sqrt{E_b})^2}{N_0} &= \ln \frac{P(s_2)}{P(s_1)} \\
\frac{4\sqrt{E_b}\,\tau}{N_0} &= \ln \frac{P(s_2)}{P(s_1)}
\end{aligned}$$

The squares τ^2 and E_b cancel in the second line. So $\tau = \dfrac{N_0}{4\sqrt{E_b}} \ln \dfrac{P(s_2)}{P(s_1)}$.

SOLUTION $\tau = \frac{0.5}{4} \ln \frac{0.75}{0.25} = 0.125 \ln 3 = 0.137$. It is positive, so it moves toward $+1$, the less likely signal. Each Q argument is the distance from a point to τ, over $\sigma = \sqrt{N_0/2} = 0.5$.

$$\begin{aligned}P_e &= 0.25\,Q\left(\frac{1-0.137}{0.5}\right) + 0.75\,Q\left(\frac{1+0.137}{0.5}\right)\\ &= 0.25\,Q(1.73) + 0.75\,Q(2.27)\\ &= 0.25(0.0422) + 0.75(0.0115) = 0.0192\end{aligned}$$

CHECK ML keeps $\tau = 0$ and gets $Q(1/0.5) = Q(2) = 0.0228$. MAP is about 16% better.

COMMON ERROR

Use $\ln\big(P(s_2)/P(s_1)\big)$, not its inverse. The inverse gives $\tau = -0.137$.

The figure below marks the threshold and the two error areas of Example 4.3.

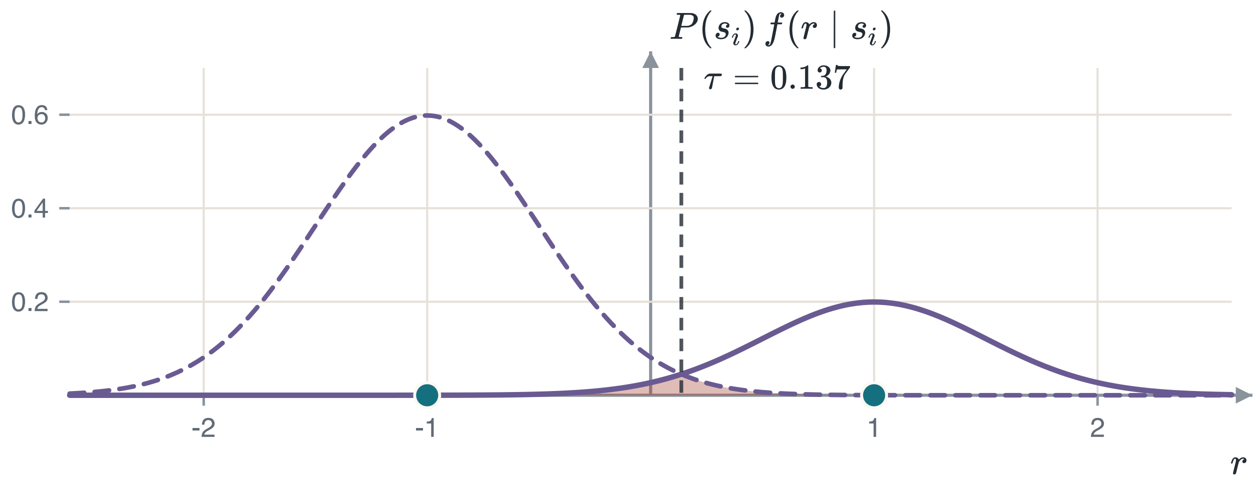


FIGURE 4.20 Example 4.3. Solid: $0.25\,f(r \mid s_1)$ at $+1$. Dashed: $0.75\,f(r \mid s_2)$ at -1. The two red areas are the two ways to be wrong, and their sum is $P_e = 0.0192$.

EXAMPLE 4.4 — A RECEIVER FOR FOUR WAVEFORMS

GIVEN Four equally likely waveforms on $[0,2)$. $s_1(t) = 2$ on $[0,1)$ and $s_2(t) = 1$ on $[0,2)$. $s_3(t)$ is -1 on $[0,1)$ and 1 on $[1,2)$, and $s_4(t) = -1$ on $[0,2)$. $\mathbf{r} = (1.2, 0.3)$ arrives.

FIND The receiver and its decision for $\mathbf{r}$.

METHOD Every waveform is constant on each half, so two unit pulses form an orthonormal basis: $\psi_1 = 1$ on $[0,1)$ and $\psi_2 = 1$ on $[1,2)$. Read each point from the heights on the two halves.

$$\mathbf{s}_1 = (2,0), \quad \mathbf{s}_2 = (1,1), \quad \mathbf{s}_3 = (-1,1), \quad \mathbf{s}_4 = (-1,-1)$$

The energies $E_i = \|\mathbf{s}_i\|^2$ are $4, 2, 2, 2$. They differ, so the metric keeps $-E_i/2$.

SOLUTION With equal priors the metric is $\mathbf{r}\cdot\mathbf{s}_i - E_i/2$.

$$\begin{aligned}\mathbf{s}_1 &: \quad 1.2(2) + 0.3(0) - 2 = 0.4\\ \mathbf{s}_2 &: \quad 1.2(1) + 0.3(1) - 1 = 0.5\\ \mathbf{s}_3 &: \quad 1.2(-1) + 0.3(1) - 1 = -1.9\\ \mathbf{s}_4 &: \quad 1.2(-1) + 0.3(-1) - 1 = -2.5\end{aligned}$$

The largest is 0.5, so $\hat{s} = s_2$.

CHECK

$\|\mathbf{r} - \mathbf{s}_1\|^2 = 0.8^2 + 0.3^2 = 0.73$ and $\|\mathbf{r} - \mathbf{s}_2\|^2 = 0.2^2 + 0.7^2 = 0.53$. So $\mathbf{s}_2$ is also the nearest point, and $\mathbf{r}$ lies in its region.

ⓧ COMMON ERROR

Use $\mathbf{r} \cdot \mathbf{s}_i - E_i/2$, not $\mathbf{r} \cdot \mathbf{s}_i$, if the energies differ. Correlation alone picks s_1.

The figure below draws the waveforms, the basis and the regions of Example 4.4.

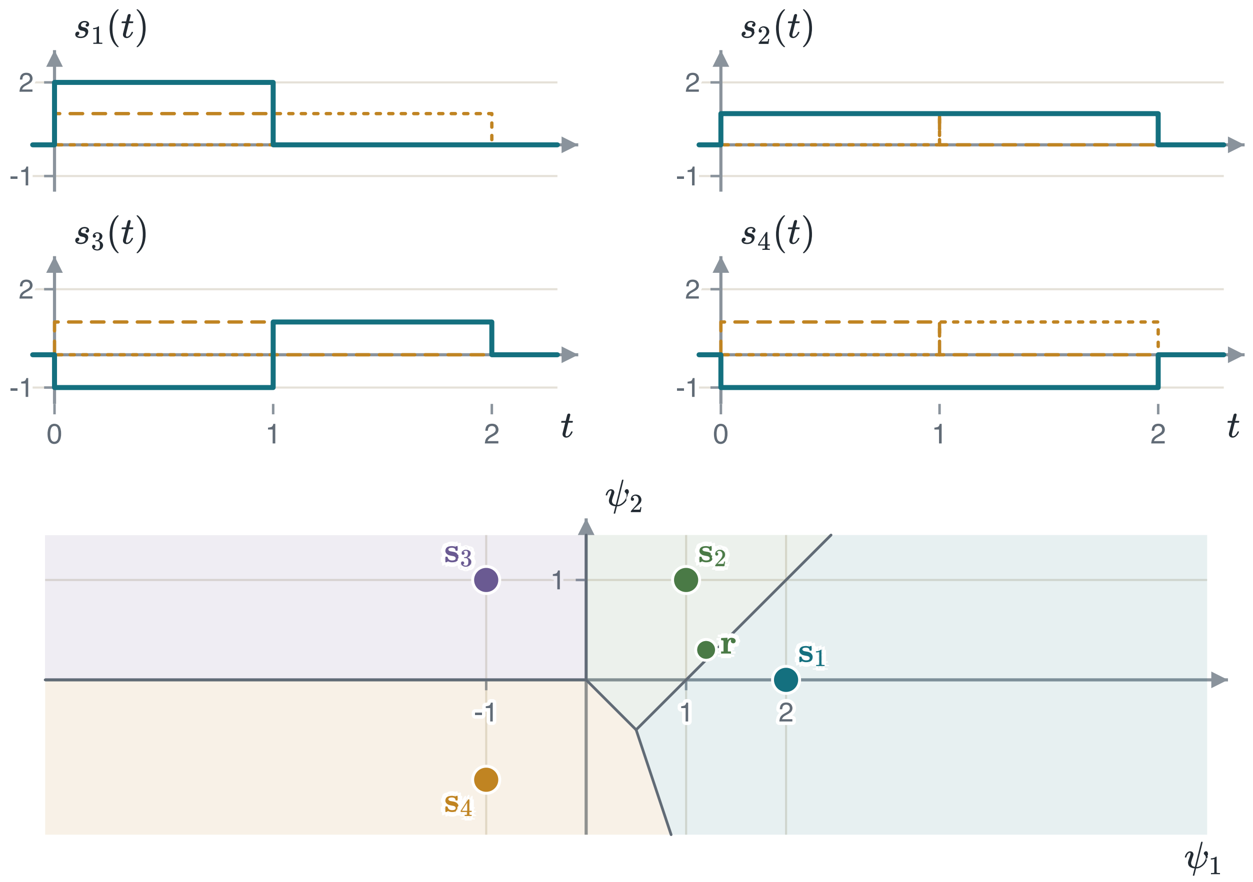


FIGURE 4.21 Example 4.4. Top: the four waveforms, with the two unit pulses dashed in amber. Bottom: the four points, the regions of the minimum-distance receiver, and $\mathbf{r} = (1.2, 0.3)$ in the region of $\mathbf{s}_2$.

Decision rules around us

Four everyday receivers use the MAP threshold, the energy term or the correlation receiver.

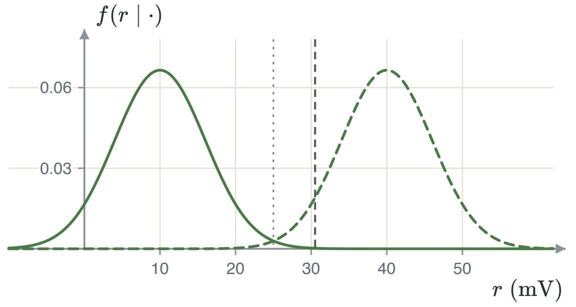


FIGURE 4.22 A smoke alarm reads 10 mV in clean air and 40 mV in smoke, with $\sigma = 6$ mV. Smoke is rare, $P = 0.01$, so the threshold moves from 25 to 30.5 mV.

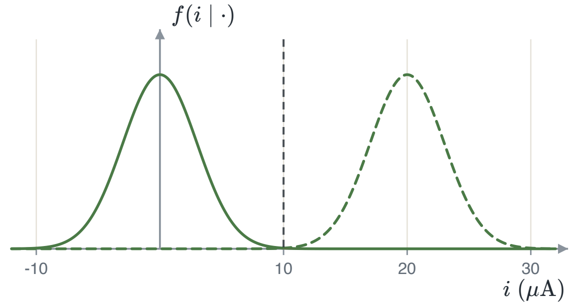


FIGURE 4.23 An optical receiver sees 0 or 20 μA. The energy term puts the threshold at 10 μA. With $\sigma = 3$ μA, $P_e = Q(10/3) = 4.3 \times 10^{-4}$.

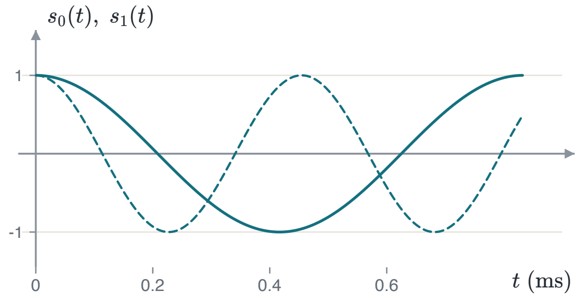


FIGURE 4.24 Caller ID sends 1200 Hz for a 1 and 2200 Hz for a 0. The receiver correlates with both tones over $T = 0.833$ ms: Method II.

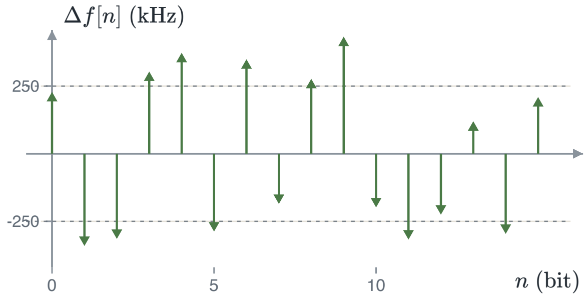


FIGURE 4.25 Bluetooth Low Energy shifts the carrier by about ± 250 kHz. With equal priors the receiver decides by the sign of each sample.

> ⓘ PRIORS MOVE THE THRESHOLD
>
> A rare event gets a smaller region. The threshold moves toward the less likely signal.

> ⓘ THE ENERGY TERM
>
> With unequal energies the metric keeps $-E_i/2$. For on-off keying that puts the threshold at half the on level.

> ⚠ UNKNOWN PRIORS
>
> When the priors are not known, the receiver uses ML. It is optimal only when the priors are equal.

4.3 Decision regions

Regions with equal priors

Group the observations that give the same decision. The space splits into M **decision regions**, one for each signal.

Start with two points. The points $\mathbf{r}$ equally far from $\mathbf{s}_i$ and $\mathbf{s}_j$ form a line, the **perpendicular bisector** of the segment between them. Expanding both squared distances shows why.

$$\|\mathbf{r} - \mathbf{s}_i\|^2 = \|\mathbf{r} - \mathbf{s}_j\|^2$$
$$-2\,\mathbf{r} \cdot \mathbf{s}_i + E_i = -2\,\mathbf{r} \cdot \mathbf{s}_j + E_j$$
$$2\,\mathbf{r} \cdot (\mathbf{s}_j - \mathbf{s}_i) = E_j - E_i$$

The $\|\mathbf{r}\|^2$ terms cancel, and the last line is linear in $\mathbf{r}$. So the boundary is a straight line, at right angles to $\mathbf{s}_j - \mathbf{s}_i$.

A REGION

$$R_i = \{\mathbf{r} : \ \|\mathbf{r} - \mathbf{s}_i\| < \|\mathbf{r} - \mathbf{s}_j\| \text{ for all } j \neq i\}$$

Each region is cut out by bisectors, so it is a convex polygon.

⚠ UNBOUNDED REGIONS

Outer points have regions that run off to infinity. Noise that pushes an outer point outward causes no error.

The figure below takes five equally likely points: the corners $(\pm 1.2, \pm 1.2)$ and the centre.

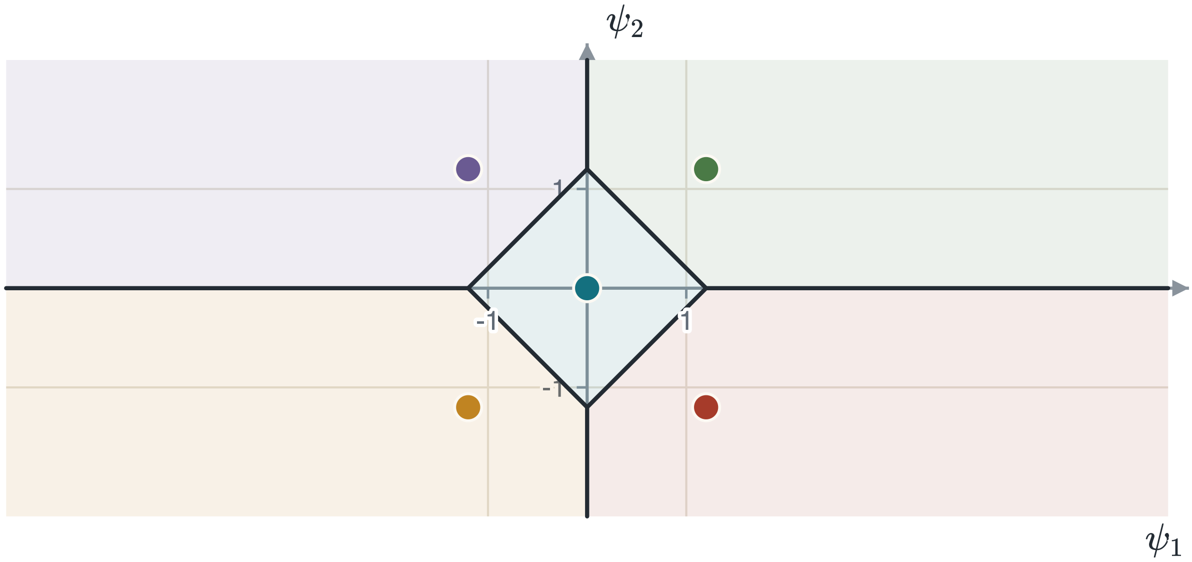


FIGURE 4.26 Five equally likely points and their regions. The centre region is the square $|r_1| + |r_2| < 1.2$, turned by $45°$. The four corner regions are unbounded.

The centre region has four straight faces. The bisector between $(0, 0)$ and $(1.2, 1.2)$ is $r_1 + r_2 = 1.2$. The other three corners give the other three faces.

Regions with unequal priors

With unequal priors the boundary between $\mathbf{s}_i$ and $\mathbf{s}_j$ is where the two MAP scores are equal.

A BOUNDARY

$$\|\mathbf{r} - \mathbf{s}_i\|^2 - N_0 \ln P(\mathbf{s}_i) = \|\mathbf{r} - \mathbf{s}_j\|^2 - N_0 \ln P(\mathbf{s}_j)$$

The $\|\mathbf{r}\|^2$ terms still cancel. The boundary is still a straight line, parallel to the bisector.

To find where the line sits, measure along the segment from $\mathbf{s}_i$ to $\mathbf{s}_j$, which is d long. Put the boundary a distance x from $\mathbf{s}_i$.

$$x^2 - N_0 \ln P(\mathbf{s}_i) = (d-x)^2 - N_0 \ln P(\mathbf{s}_j)$$
$$2dx - d^2 = N_0 \ln \frac{P(\mathbf{s}_i)}{P(\mathbf{s}_j)}$$

Solve for x and call the result μ_i.

THE SHIFT

$$\mu_i = \frac{d}{2} + \frac{N_0}{2d} \ln \frac{P(\mathbf{s}_i)}{P(\mathbf{s}_j)}$$

μ_i is the distance from $\mathbf{s}_i$ to its boundary with $\mathbf{s}_j$. A likely point pushes the line away from itself, toward the less likely point.

For example, take $d = 2$, $N_0 = 1$, $P(\mathbf{s}_i) = 0.8$ and $P(\mathbf{s}_j) = 0.2$. Then $\mu_i = 1 + \frac{1}{4} \ln 4 = 1 + 0.35 = 1.35$. The likely point keeps the larger share.

The figure below gives each of three points a prior, with $N_0 = 1$.

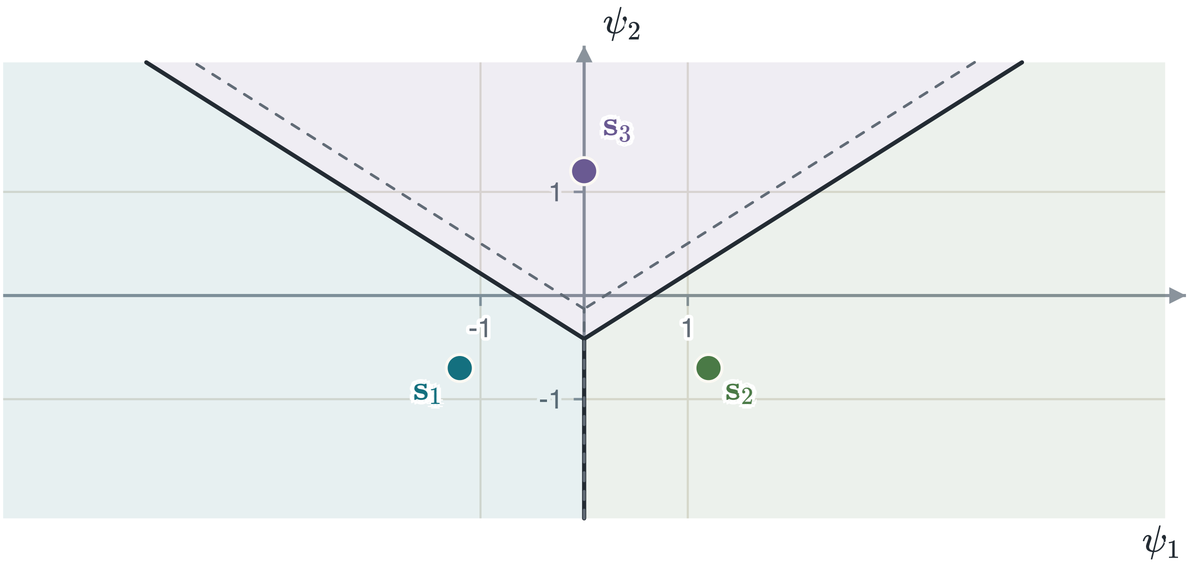


FIGURE 4.27 Three points with $N_0 = 1$, drawn for $P(\mathbf{s}_3) = 0.6$ and $P(\mathbf{s}_1) = P(\mathbf{s}_2) = 0.2$. Solid lines are the MAP boundaries, dashed lines the equal-prior ones. The likely $\mathbf{s}_3$ takes more of the plane.

Binary error probability

Two equally likely points a distance d apart give the simplest error probability. The boundary is the midpoint, $d/2$ from each point.

Only the noise along the line through the two points can cause an error. That component has variance $N_0/2$, and an error needs more than $d/2$ of it.

ALONG THE LINE

$$P_e = P\left(n > \frac{d}{2}\right) = Q\left(\frac{d/2}{\sqrt{N_0/2}}\right)$$
$$= Q\left(\sqrt{\frac{d^2}{2N_0}}\right)$$

Here $Q(x) = \frac{1}{2}\,\mathrm{erfc}(x/\sqrt{2})$ is the Gaussian tail. The last step squares the argument: $(d/2)^2/(N_0/2) = d^2/2N_0$.

Put the distance of each special case into $d^2/2N_0$.

TWO SPECIAL CASES

$$\text{antipodal, } d = 2\sqrt{E_b}: \qquad \frac{d^2}{2N_0} = \frac{2E_b}{N_0}, \qquad P_e = Q\big(\sqrt{2E_b/N_0}\big)$$
$$\text{orthogonal or on-off, } d = \sqrt{2E_b}: \qquad \frac{d^2}{2N_0} = \frac{E_b}{N_0}, \qquad P_e = Q\big(\sqrt{E_b/N_0}\big)$$

For example, $d = 2$ and $N_0 = 0.5$ give $d^2/2N_0 = 4/1 = 4$, so $P_e = Q(2) = 0.0228$.

The figure below draws the two densities along the line for $d = 2$ and $N_0 = 1$.

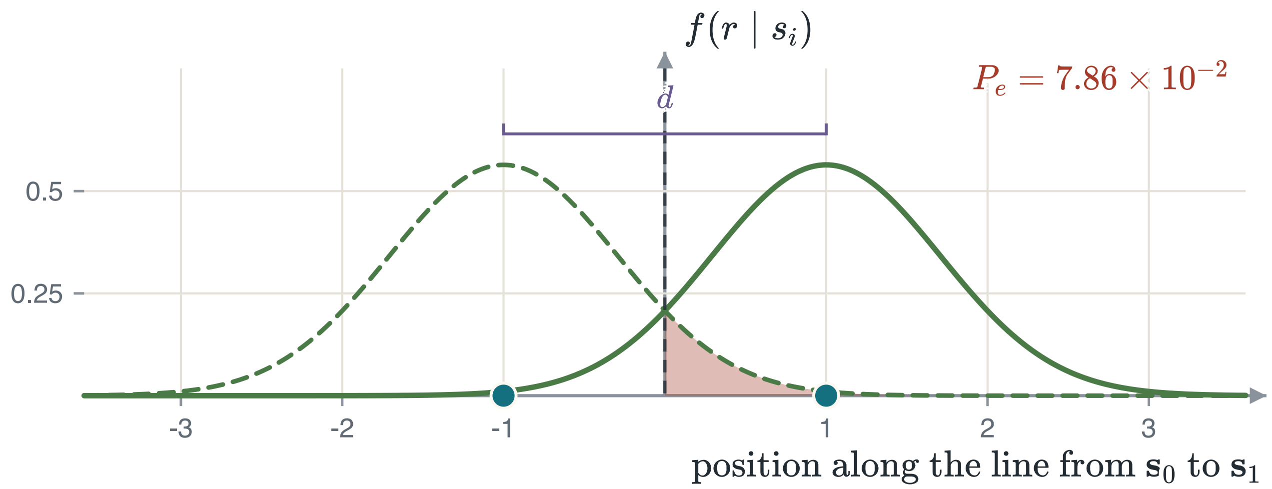


FIGURE 4.28 Two equally likely points $d = 2$ apart, with $N_0 = 1$. The red tail of the left density past the midpoint is $P_e = Q(\sqrt{2}) = 7.86 \times 10^{-2}$.

Binary error with unequal priors

With unequal priors the MAP boundary moves off the midpoint. Put s_0 and s_1 a distance d apart, with the boundary a distance μ from s_1.

THE BOUNDARY

$$\mu = \frac{d}{2} + \frac{N_0}{2d} \ln \frac{P(s_1)}{P(s_0)}$$

This is the shift μ_i above, with s_1 in the place of $\mathbf{s}_i$.

If s_0 is sent, an error needs noise past $d - \mu$. If s_1 is sent, it needs noise past μ. Weight each tail by its prior.

TWO TAILS

$$P_e = P(s_0)\, Q\!\left(\frac{d-\mu}{\sqrt{N_0/2}}\right) + P(s_1)\, Q\!\left(\frac{\mu}{\sqrt{N_0/2}}\right)$$

Equal priors give $\mu = d/2$ and the formula for two equally likely points.

For example, take $d = 2$, $N_0 = 0.5$ and $P(s_1) = 0.9$. The boundary and the error follow in two steps.

$$\begin{aligned}
\mu &= 1 + \frac{0.5}{4} \ln \frac{0.9}{0.1} = 1 + 0.125(2.197) = 1.27 \\
P_e &= 0.1\, Q\!\left(\frac{2-1.27}{0.5}\right) + 0.9\, Q\!\left(\frac{1.27}{0.5}\right) \\
&= 0.1\, Q(1.45) + 0.9\, Q(2.55) = 0.0122
\end{aligned}$$

The ML receiver keeps the midpoint and gets $Q(2) = 0.0228$. So the MAP receiver has the smaller P_e.

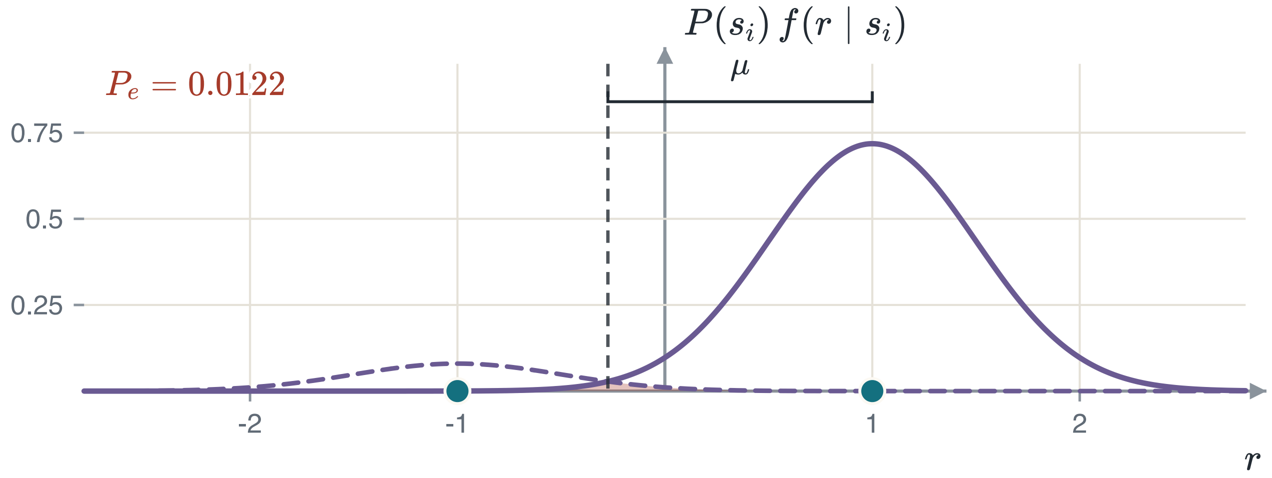


FIGURE 4.29 $s_0 = -1$ and $s_1 = +1$, so $d = 2$, with $N_0 = 0.5$, drawn for $P(s_1) = 0.9$. The boundary sits $\mu = 1.27$ from s_1, and the two red tails add up to $P_e = 0.0122$.

Decision regions around us

Four everyday systems decide in regions cut by bisectors.

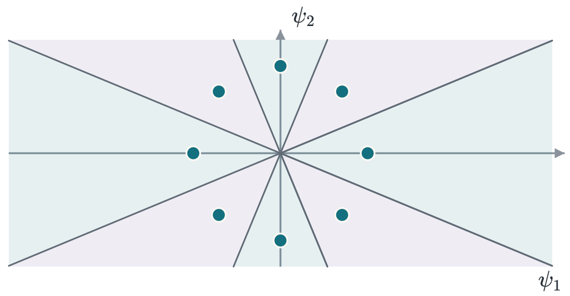


FIGURE 4.30 Satellite television can use 8PSK. Its regions are eight 45° sectors, so the receiver decides by the phase of $\mathbf{r}$ alone.

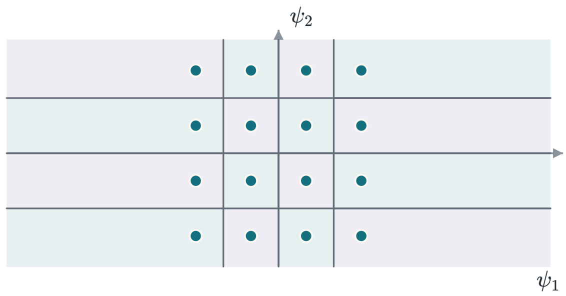


FIGURE 4.31 Wi-Fi 16-QAM at unit average energy has square regions of side $d_{\min} = 2/\sqrt{10} = 0.632$. The 12 outer ones are unbounded.

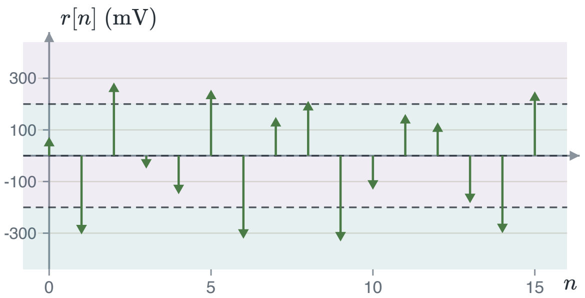


FIGURE 4.32 PAM4 in data-centre links sends ± 100 and ± 300 mV. The thresholds 0 and ± 200 mV cut the axis into four intervals.

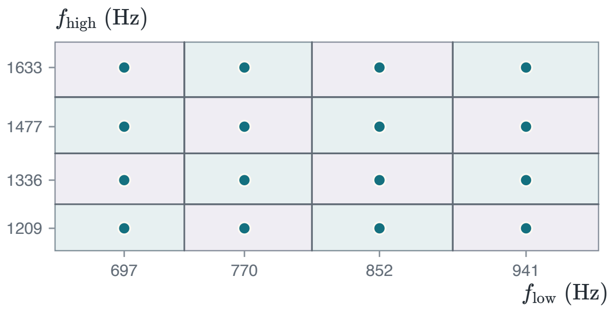


FIGURE 4.33 A phone key sends one low and one high tone. The receiver decides in rectangles with edges halfway between neighbouring tones.

> ⓘ NEAREST POINT
>
> In each system the receiver picks the signal point nearest to $\mathbf{r}$. The regions are cut by perpendicular bisectors.

> ⓘ OUTER REGIONS
>
> Points on the edge of a constellation have unbounded regions. Noise can push them outward without an error.

> ⚠ SCALE
>
> The regions hold only if the receiver knows the scale of the points. A gain error moves every point and every boundary.

4.4 Error probability and the union bound

The exact error probability

The exact error probability adds, over the signals, the part of each Gaussian bell that falls outside its region.

EXACT ERROR

$$P_e = \sum_{i=1}^{M} P(\mathbf{s}_i) \int_{\mathbf{r} \notin R_i} f(\mathbf{r} \mid \mathbf{s}_i)\, d\mathbf{r}$$

Each term is the part of a Gaussian bell that falls outside a polygon.

⚠ NO CLOSED FORM

For most polygons this integral has no formula. It needs numerical integration or a simulation.

A **simulation** sends many symbols, adds noise and counts the wrong decisions. About 100 errors give a usable estimate. At $P_e = 10^{-4}$ that takes about $100/10^{-4} = 10^6$ symbols.

Sometimes the integral is easy. In the five-point set of Section 4.3, send the centre with $\sigma = 0.4$. Its region is the square $|r_1| + |r_2| < 1.2$, turned by $45°$.

Turn the axes with the square. Each new noise component still has variance σ^2. The region becomes $|u_1| < a$ and $|u_2| < a$, with half-width $a = 1.2/\sqrt{2} = 0.849$.

$$\begin{aligned} P(\text{error} \mid \text{centre}) &= 1 - \Big(1 - 2Q\big(a/\sigma\big)\Big)^2 \\ &= 1 - \Big(1 - 2Q(2.12)\Big)^2 = 0.067 \end{aligned}$$

The figure below counts the draws that land outside the centre region.

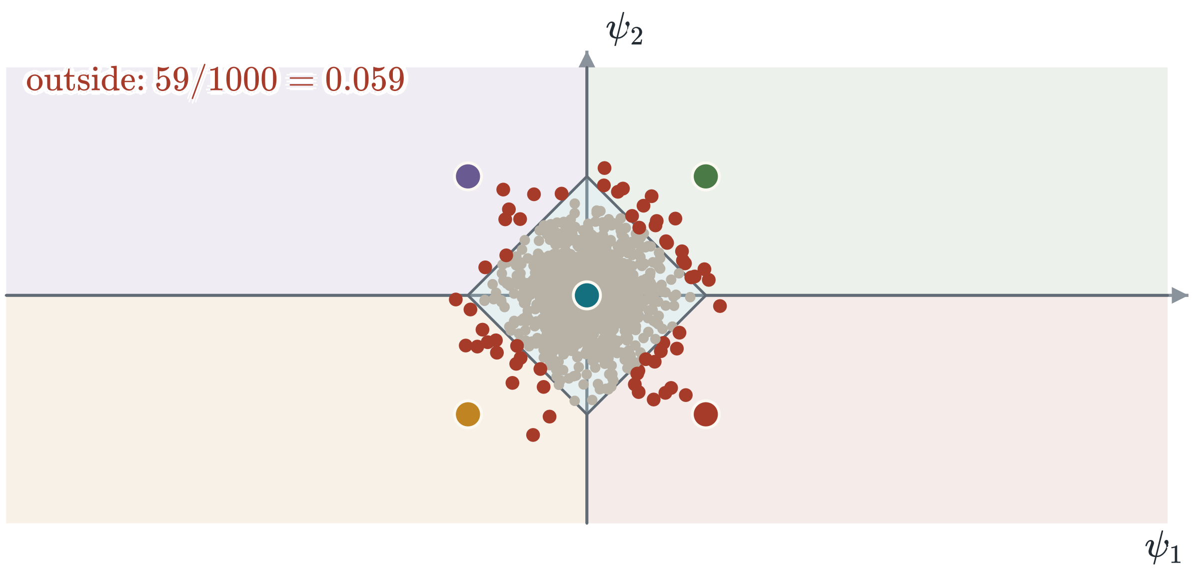


FIGURE 4.34 The centre point of the five-point set is sent with $\sigma = 0.4$, drawn for 1000 draws. The red draws fall outside its region, and their fraction settles near the exact 0.067.

The union bound

The union bound avoids the integral. Suppose $\mathbf{s}_i$ was sent, and let A_{ij} be the event that $\mathbf{r}$ is nearer $\mathbf{s}_j$ than $\mathbf{s}_i$.

An error happens when at least one A_{ij} happens, so the error is the union of these events. The probability of a union is at most the sum of the probabilities.

PAIRWISE EVENTS

$$P(\text{error} \mid \mathbf{s}_i) \le \sum_{j \ne i} P(A_{ij})$$

$$P(A_{ij}) = Q\left(\sqrt{\frac{d_{ij}^2}{2N_0}}\right)$$

Each A_{ij} is a binary error between two points at distance d_{ij}. So its probability is the binary result of Section 4.3.

ⓘ OVERLAPS COUNTED TWICE

The events overlap. The sum counts each overlap more than once, so it can only be larger than the true probability.

Average over the signals with equal priors $1/M$. This gives the union bound, which needs only the distances between the points.

UNION BOUND

$$P_e \le \frac{1}{M} \sum_{i=1}^{M} \sum_{j \ne i} Q\left(\sqrt{\frac{d_{ij}^2}{2N_0}}\right)$$

The figure below draws the three half-planes for $\mathbf{s}_1 = (1, 1)$ of the square.

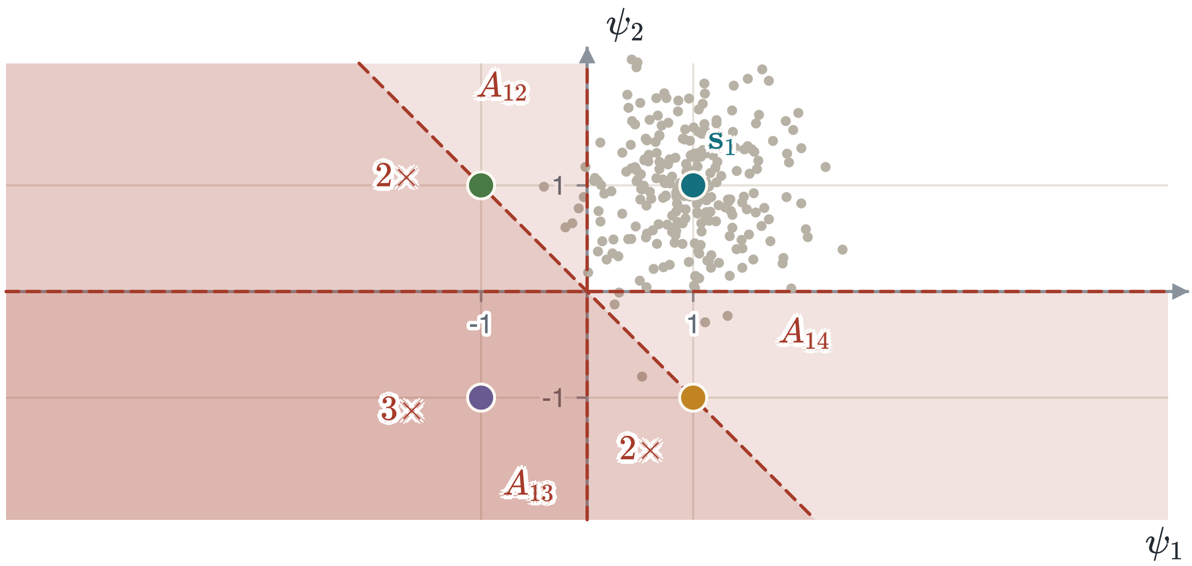


FIGURE 4.35 $\mathbf{s}_1 = (1, 1)$ is sent, and the grey dots are received points. Each red half-plane A_{1j} is where $\mathbf{r}$ is nearer $\mathbf{s}_j$. The labels mark the parts the sum counts two or three times.

For $\mathbf{s}_1$ the bound has $M-1=3$ terms. Two are for the points at $d=2$, and one is for the diagonal point at $2\sqrt{2}$.

The minimum-distance bound

Every d_{ij} is at least the smallest distance $d_{\min}$, and Q falls as its argument grows. So each term is at most the term at $d_{\min}$.

MINIMUM-DISTANCE BOUND

$$P_e \le (M-1)\,Q\left(\sqrt{\frac{d_{\min}^2}{2N_0}}\right)$$

Each of the M signals has $M-1$ terms, each at most $Q\big(\sqrt{d_{\min}^2/2N_0}\big)$. The average over the M signals leaves $M-1$ of them.

ⓘ ONE NUMBER

The bound needs only M and the smallest distance. It is quick, and loose when few pairs sit at $d_{\min}$.

8-PSK puts $M=8$ points on a circle of radius $\sqrt{E_s}$. The point j steps away sits at $d_j = 2\sqrt{E_s}\,\sin(\pi j/8)$. With $E_s=1$, $d_{\min} = 2\sin(\pi/8) = 2(0.383) = 0.77$.

Only two of the seven other points sit at $d_{\min}$. So the bound, which puts all seven there, is loose.

The figure below compares it with the union bound for 8-PSK. Each term uses $d_j^2/2N_0 = (2E_s/N_0)\sin^2(\pi j/8)$.

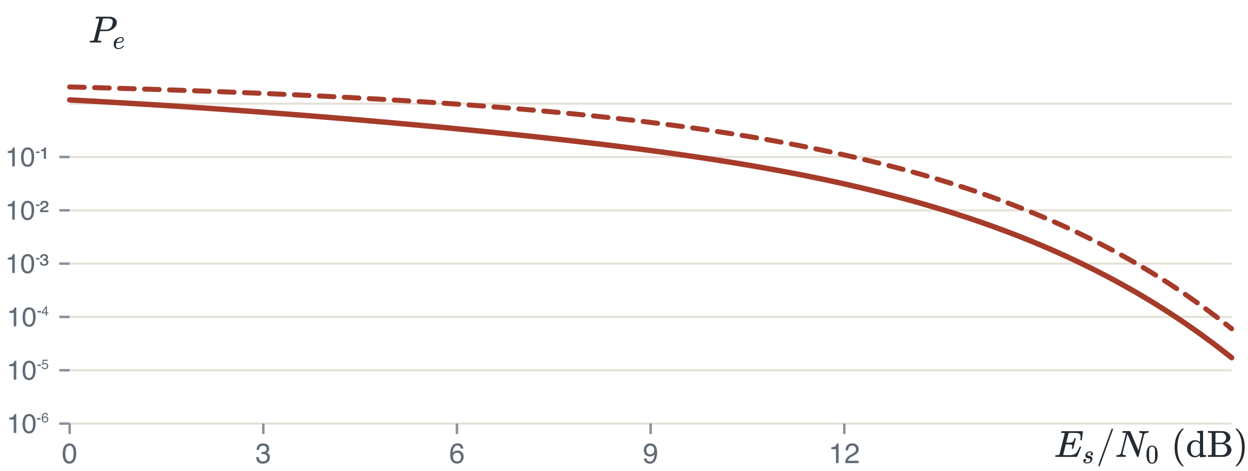


FIGURE 4.36 8-PSK. Solid: the union bound, the sum of all seven pairwise terms. Dashed: the minimum-distance bound, with every distance replaced by $d_{\min}$. The dashed curve stays above.

An exponential bound

The function $Q(x)$ has no closed form. A plain exponential sits above it for every $x \ge 0$.

EXPONENTIAL BOUND

$$Q(x) \le \tfrac{1}{2}e^{-x^2/2}, \qquad x \ge 0$$

Put $x^2 = d_{\min}^2/2N_0$ into the minimum-distance bound. Then $x^2/2 = d_{\min}^2/4N_0$.

APPLIED TO THE BOUND

$$P_e \le \frac{M-1}{2}\, e^{-d_{\min}^2/4N_0}$$

The error falls exponentially in $d_{\min}^2/N_0$.

For example, at $x = 3$ the bound is $\frac{1}{2}e^{-4.5} = 5.6 \times 10^{-3}$ against $Q(3) = 1.35 \times 10^{-3}$. It is about 4.1 times larger. The figure below draws both on a logarithmic axis.

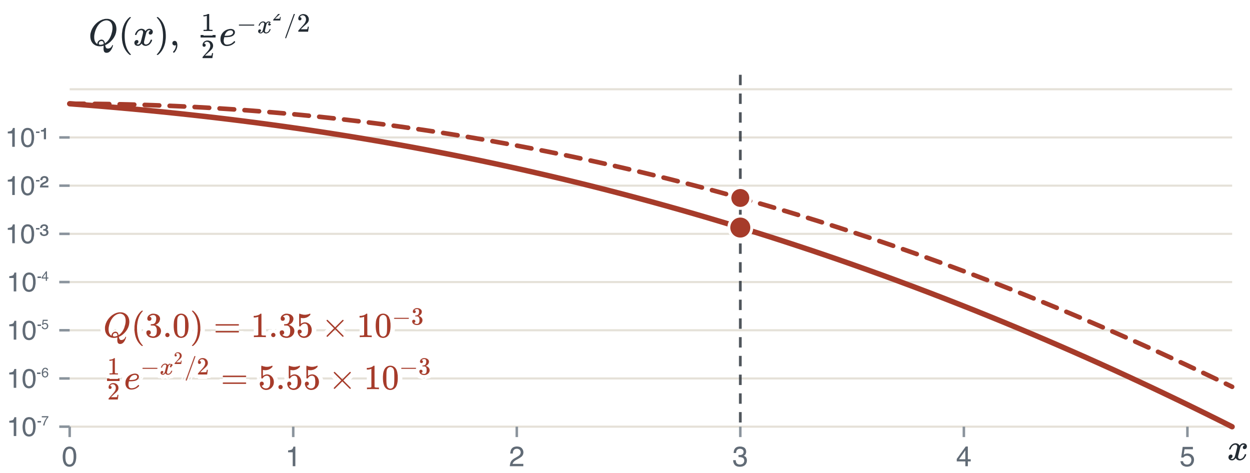


FIGURE 4.37 Solid: $Q(x)$. Dashed: $\frac{1}{2}e^{-x^2/2}$, marked at $x = 3$. The bound stays above $Q(x)$, and both fall with the same exponent.

The intelligent union bound

To leave R_i, the point $\mathbf{r}$ must cross one of the faces of R_i. So keep the terms for the points that share a face with R_i, a set written F_i. Drop the others.

INTELLIGENT UNION BOUND

$$P(\text{error} \mid \mathbf{s}_i) \le \sum_{j \in F_i} Q\left(\sqrt{\frac{d_{ij}^2}{2N_0}}\right)$$

The bound is tighter than the general union bound, and still a bound.

In the square, the region of $\mathbf{s}_1$ is the first quadrant. The figure below marks its two faces, shared with $\mathbf{s}_2$ and $\mathbf{s}_4$. So the bound keeps two terms and drops the term for $\mathbf{s}_3$.

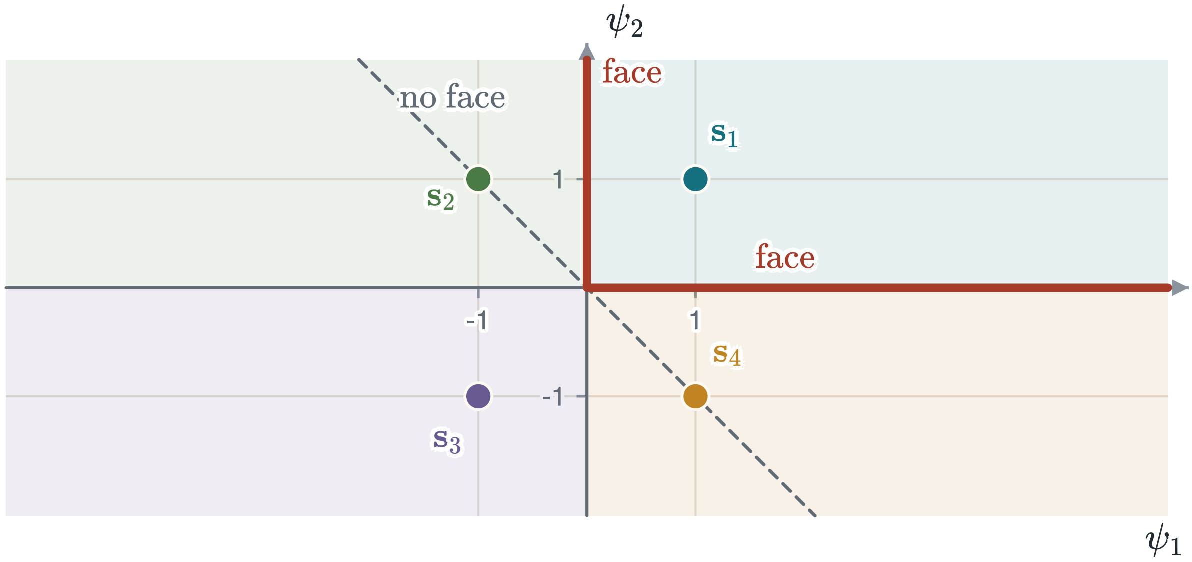


FIGURE 4.38 The region of $\mathbf{s}_1$ is the first quadrant. Its two faces, in red, are shared with $\mathbf{s}_2$ and $\mathbf{s}_4$. The bisector with $\mathbf{s}_3$, dashed, only touches the corner.

Nearest neighbours

At high SNR the terms at $d_{\min}$ are far larger than the others. Keeping only those terms gives the nearest-neighbour form.

NEAREST NEIGHBOURS

$$P_e \approx \bar{N}_{\min}\, Q\left(\sqrt{\frac{d_{\min}^2}{2N_0}}\right)$$

$\bar{N}_{\min}$ is the average number of neighbours at $d_{\min}$, taken over the M points.

⚠ AN APPROXIMATION

It keeps only part of the union bound, so it is not a bound. It can sit below the exact P_e.

In QPSK on the square $(\pm 1, \pm 1)$, each point has two neighbours at $d_{\min} = 2$. The diagonal point is at $2\sqrt{2}$, so $\bar{N}_{\min} = 2$.

In 16-QAM on the grid $\pm 1, \pm 3$, $d_{\min} = 2$. The four corners have two neighbours at $d_{\min}$, the eight edge points three, and the four inner points four.

$$\bar{N}_{\min} = \frac{4(2) + 8(3) + 4(4)}{16} = \frac{48}{16} = 3$$

The figure below joins every pair of neighbours in 16-QAM.

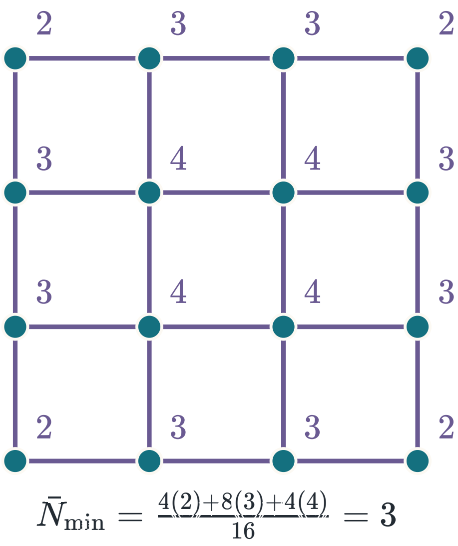


FIGURE 4.39 16-QAM on the grid $\pm1, \pm3$. Each line joins two neighbours at $d_{\min} = 2$, and each number counts the neighbours of its point.

EXAMPLE 4.5 — FOUR FORMS ON QPSK

GIVEN QPSK at $(\pm1, \pm1)$, equally likely, with $N_0 = 2/9$. So $d^2_{\min}/2N_0 = 4/(4/9) = 9$.

FIND The general union bound, the intelligent bound, the nearest-neighbour form and the minimum-distance bound. Which is largest?

METHOD From $\mathbf{s}_1 = (1, 1)$, two points are at $d = 2$, which gives $x = \sqrt{9} = 3$. One is at $2\sqrt{2}$, which gives $x = \sqrt{8/(4/9)} = \sqrt{18} = 4.24$. The region of $\mathbf{s}_1$ is a quadrant with two faces, and all four points are alike.

$$Q(3) = 1.35 \times 10^{-3}, \qquad Q(4.24) = 1.1 \times 10^{-5}$$

SOLUTION

$$\begin{aligned} \text{general} &= 2Q(3) + Q(4.24) = 2.71 \times 10^{-3} \\ \text{intelligent} = \text{nearest} &= 2Q(3) = 2.70 \times 10^{-3} \\ d_{\min} \text{ bound} &= 3Q(3) = 4.05 \times 10^{-3} \end{aligned}$$

The minimum-distance bound is the largest. It counts three terms at $Q(3)$, and the others count two plus a tiny diagonal term.

CHECK The regions are quadrants, so each noise component acts on its own. The exact value is $1 - \left(1 - Q(3)\right)^2 = 2.70 \times 10^{-3}$. The diagonal term adds only 0.4%.

COMMON ERROR

Use $\bar{N}_{\min} = 2$, not 3. The diagonal point is at $2\sqrt{2}$, not at $d_{\min} = 2$.

The figure below shows the nearest neighbours of $\mathbf{s}_1$ in Example 4.5.

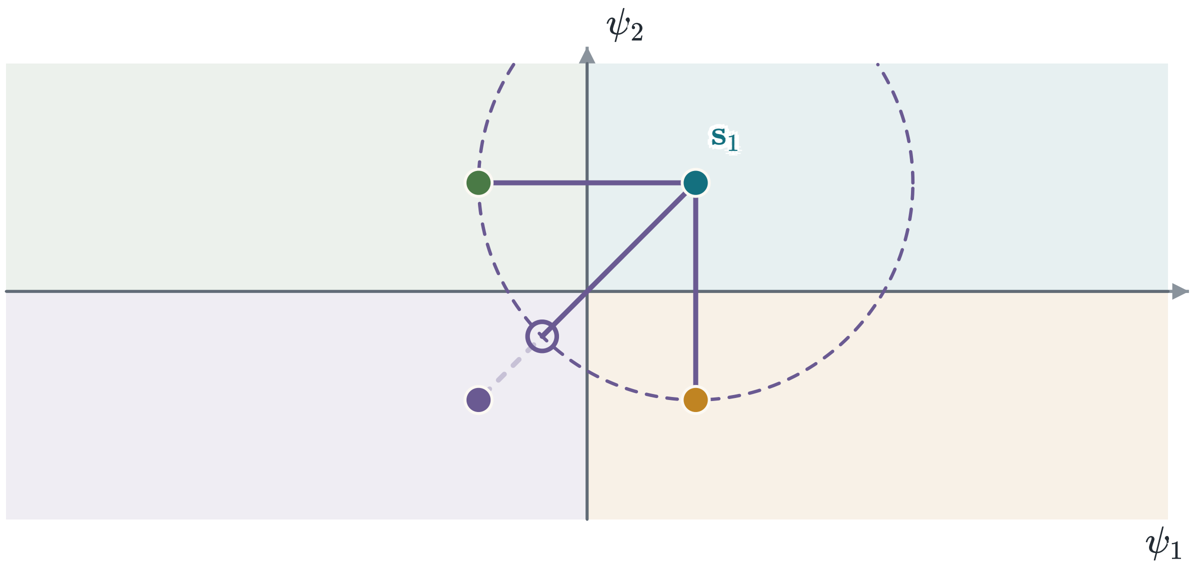


FIGURE 4.40 Example 4.5 from $\mathbf{s}_1$. The dashed circle of radius $d_{\min}$ passes through the two nearest neighbours. The minimum-distance bound moves the diagonal point in to $d_{\min}$, ringed.

EXAMPLE 4.6 — A RECTANGLE

GIVEN Four equally likely points $(\pm 1.5, \pm 1)$ with $N_0 = 0.4$.

FIND The intelligent bound and the nearest-neighbour form. Which point shares a face with $\mathbf{s}_1 = (1.5, 1)$ but is not a nearest neighbour?

METHOD From $\mathbf{s}_1$ the others are at 3, 2 and $\sqrt{13}$. The region of $\mathbf{s}_1$ is the first quadrant, and its faces are shared with the points at 3 and 2. Each Q argument is $d/\sqrt{2N_0} = d/\sqrt{0.8}$.

$$d = 2: \;\; Q(2.24) = 1.27 \times 10^{-2}, \qquad d = 3: \;\; Q(3.35) = 4.0 \times 10^{-4}$$

SOLUTION

$$\begin{aligned} \text{intelligent} &= Q(2.24) + Q(3.35) = 1.31 \times 10^{-2} \\ \text{nearest} &= \bar{N}_{\min}\, Q(2.24) = 1.27 \times 10^{-2} \end{aligned}$$

Each point has one neighbour at $d_{\min} = 2$, so $\bar{N}_{\min} = 1$. The point $(-1.5, 1)$ shares the face $r_1 = 0$ but sits at 3, more than $d_{\min}$.

CHECK The regions are quadrants, so the exact value is $1 - \big(1 - Q(3.35)\big)\big(1 - Q(2.24)\big) = 1.31 \times 10^{-2}$. The nearest-neighbour form is 3% below it, so it is not a bound.

> **COMMON ERROR**
> Use $Q(2.24) + Q(3.35)$ for both faces, not only $Q(2.24)$ for the nearest neighbour.

The figure below marks the faces and the nearest neighbour of $\mathbf{s}_1$ in Example 4.6.

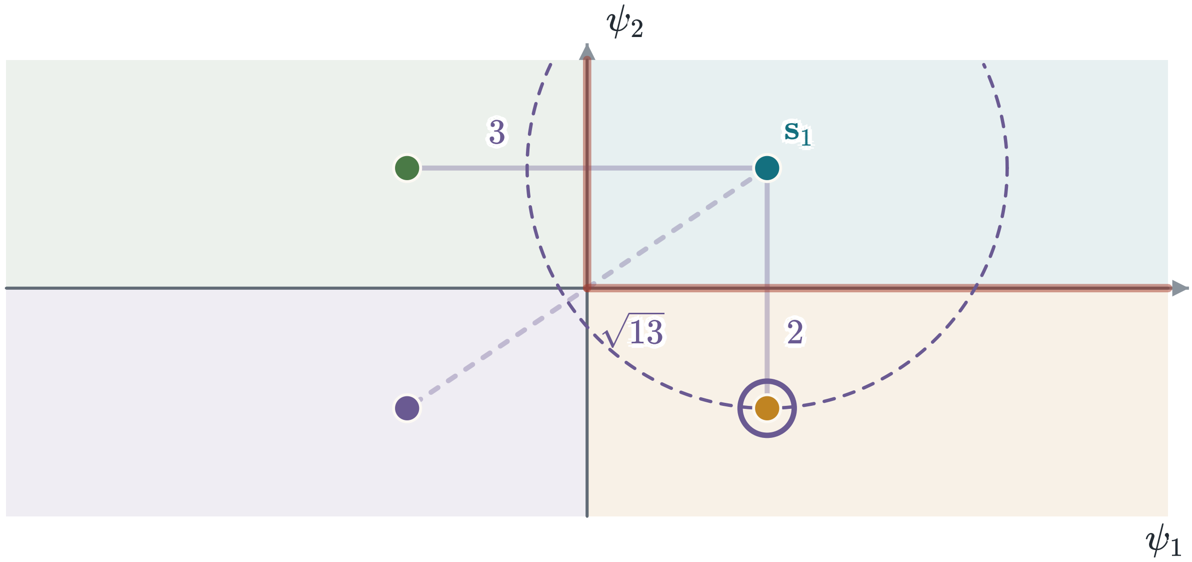


FIGURE 4.41 Example 4.6. The two red faces of R_1 are shared with the points at 3 and 2. The dashed circle of radius $d_{\min} = 2$ passes through one nearest neighbour, ringed.

Tightness of the bounds

The bounds differ most at low SNR. Many terms are then large, and they overlap.

> ⓘ LOW SNR
>
> The union bound can pass 1 and then tells nothing. For 16-QAM with $E_s = 10$ at $E_s/N_0 = 0$ dB, it adds 15 heavily overlapping terms and gives 2.79.

> ✓ HIGH SNR
>
> The terms at $d_{\min}$ dominate. The union bound, the nearest-neighbour form and the simulation meet.

> ⚠ THE MINIMUM-DISTANCE BOUND
>
> It counts all 15 other points at $d_{\min}$, five times the $\bar{N}_{\min} = 3$ of 16-QAM. So it stays five times too high.

The figure below compares the three forms with a simulation for 16-QAM.

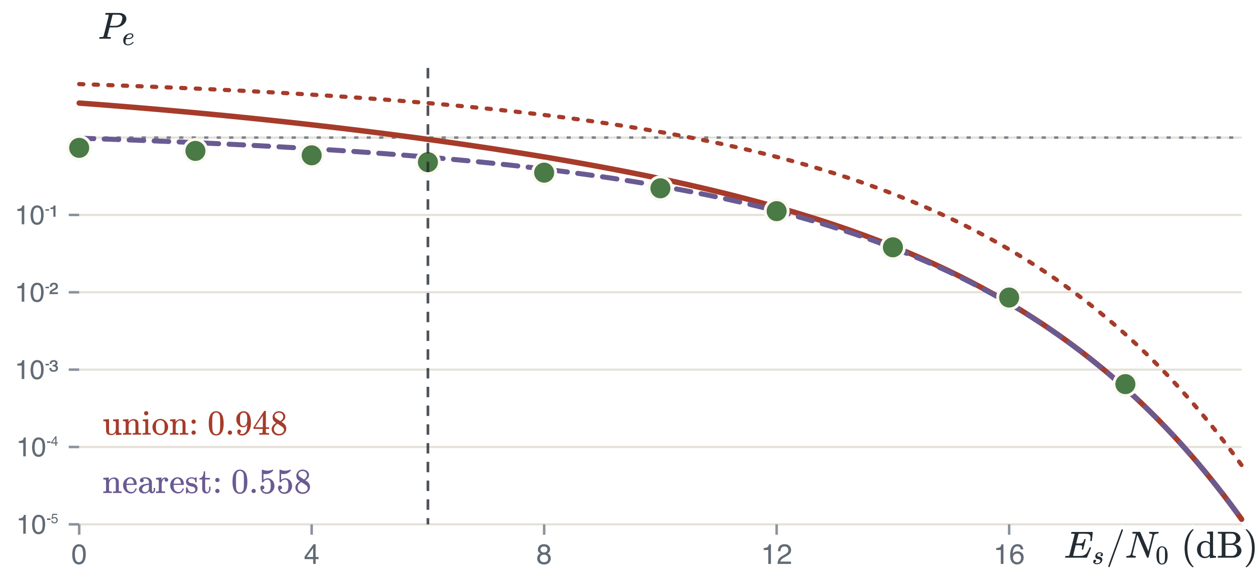


FIGURE 4.42 16-QAM with $E_s = 10$, marked at $E_s/N_0 = 6$ dB. Solid red: the union bound. Dotted red: the minimum-distance bound. Dashed violet: the nearest-neighbour form. Green dots: a simulation of $20\,000$ symbols. The dashed line at the top is $P_e = 1$.

Error probability around us

Wi-Fi, LTE and 5G use the nearest-neighbour form and the union bound, and a tester measures the error rate.

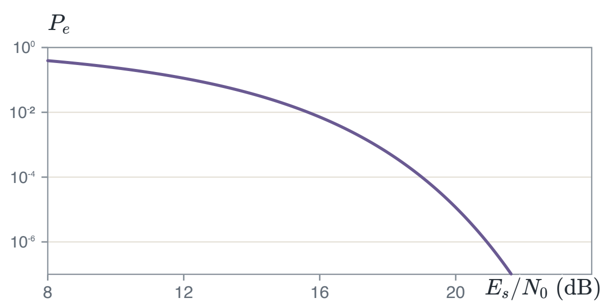


FIGURE 4.43 Wi-Fi 16-QAM has $d_{\min}^2 = 0.4E_s$ and $\bar{N}_{\min} = 3$. The nearest-neighbour form is $P_e \approx 3Q\left(\sqrt{E_s/5N_0}\right)$.

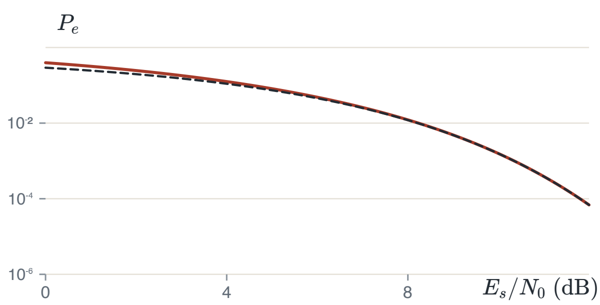


FIGURE 4.44 LTE control channels use QPSK. Solid: the union bound. Dashed: the exact P_e. They differ by less than 1% above 10 dB.

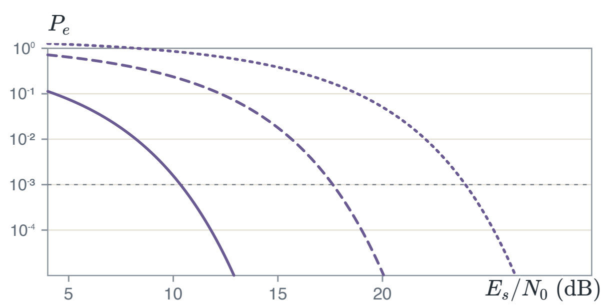


FIGURE 4.45 A 5G phone picks its modulation from the channel. For $P_e = 10^{-3}$, QPSK, 16-QAM and 64-QAM need about 10.3, 17.6 and 24.0 dB.

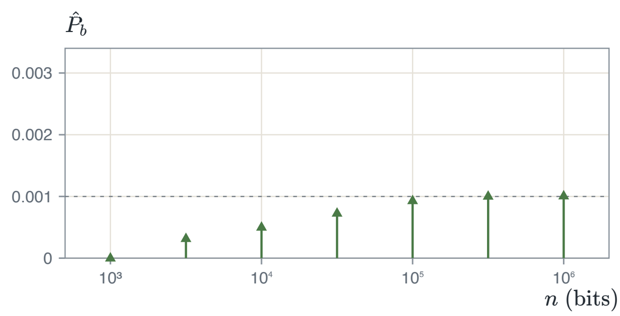


FIGURE 4.46 A bit-error-rate tester divides the errors by the bits sent. At $P_b = 10^{-3}$ the estimate settles once about 100 errors are in.

ⓘ NEAREST NEIGHBOURS FIRST

At useful error rates the terms at $d_{\min}$ dominate. Designers quote $\bar{N}_{\min} Q\left(\sqrt{d_{\min}^2/2N_0}\right)$.

ⓘ A TARGET ERROR RATE

A link fixes a target such as 10^{-3}. Each constellation then needs its own E_s/N_0.

⚠ FEW ERRORS

An error rate measured from a few errors is noisy. A tester waits for about 100 errors.

4.5 Summary

From waveform to error rate

The figure below follows one symbol through the receiver and then repeats it.

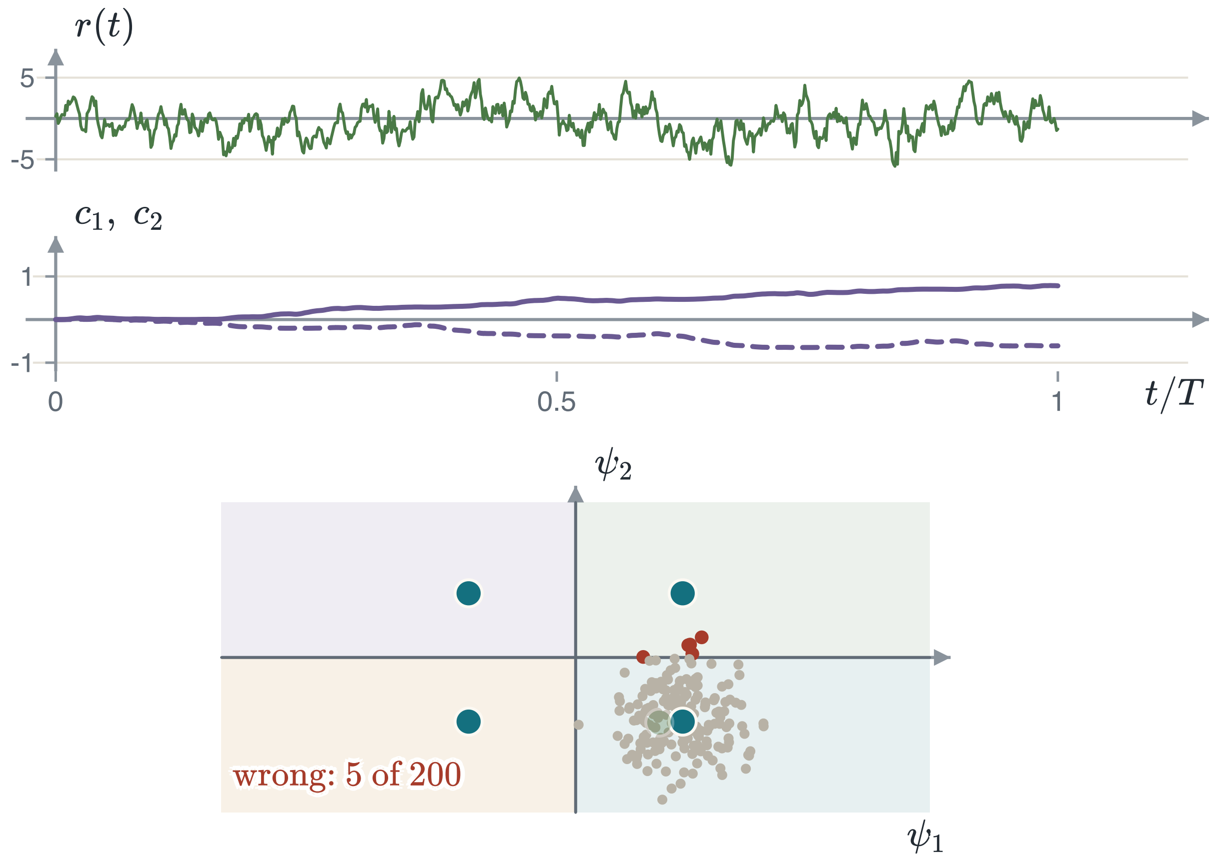


FIGURE 4.47 One symbol through the receiver: $r(t)$, the correlator outputs c_1 and c_2, and the regions of four points $(\pm1, \pm0.6)$. The 200 repeats with $\sigma = 0.3$ estimate P_e, and the wrong ones are red.

The receiver of this chapter makes four moves.

1. Correlate $r(t)$ with the basis to get $\mathbf{r}$.
2. Pick the best metric: the nearest point for equal priors.

3. Read the regions from the bisectors.
4. Bound P_e from the distances.

Each modulation of Chapter 5 is a constellation. Its error rate follows from its distances and this receiver.

For example, take QPSK with $d_{\min}^2/2N_0 = 16$. Here $\bar{N}_{\min} = 2$ and $Q(4) = 3.17 \times 10^{-5}$, so the nearest-neighbour form gives $P_e \approx 6.3 \times 10^{-5}$.

Quick check

1 A set of $M = 32$ signals. How many bits does each symbol carry?
Answer: 5 bits, since $\log_2 32 = 5$.

2 $N_0 = 0.4$. What is the noise variance of each correlator output?
Answer: $N_0/2 = 0.2$.

3 $\mathbf{r} = (0.2, 0.9)$, with $(0, 1)$ and $(1, 0)$ equally likely. Which point does the receiver pick?
Answer: $(0, 1)$. The squared distances are 0.05 and 1.45, and the smaller wins.

4 $P(\mathbf{s}_1)$ rises. Which way does the boundary between $\mathbf{s}_1$ and $\mathbf{s}_2$ move?
Answer: Away from $\mathbf{s}_1$. A likely point gets a larger region, so the line moves toward $\mathbf{s}_2$.

5 Two equally likely points, $d = 2$ and $N_0 = 2$. What is P_e?
Answer: $Q(1)$, since $d^2/2N_0 = 4/4 = 1$.

6 A set of $M = 8$ points. For one sent point, how many terms does the general union bound have?
Answer: 7 terms, one for each other point, $M - 1 = 7$.

Summary of results

TABLE 4.1 Summary of Chapter 4: the optimal receiver in additive white Gaussian noise.

RESULT	STATEMENT	ANCHOR
Bits and symbols	$k = \log_2 M$, so $R_b = kR_s$ and $T_b = T/k$	PS CH8.4
Correlator bank	$\mathbf{r} = \mathbf{s}_i + \mathbf{n}$, with $r_j = \int_0^T r(t)\, \psi_j(t)\, dt$	PS CH8.4.1
Matched filter	the same number at the sampling time $t = T$, with impulse response $\psi_j(T - t)$	PS CH8.4.1
Noise outside the signal space	does not depend on the signal and adds the same $\|n'\|^2$ to every distance	PS CH8.4.1
Noise components	independent Gaussians with mean 0 and variance $N_0/2$. The cloud is round	PS CH8.4.1
MAP and ML	MAP maximises $P(\mathbf{s}_i)\, f(\mathbf{r} \mid \mathbf{s}_i)$. With equal priors it is ML	PS CH8.4.1
Minimum distance	ML in Gaussian noise. MAP subtracts $N_0 \ln P(\mathbf{s}_i)$ from each squared distance	PS CH8.4.1
Correlation metric	$\mathbf{r} \cdot \mathbf{s}_i - E_i/2 + \frac{N_0}{2} \ln P(\mathbf{s}_i)$. Keep the largest	PS CH8.4.1
Decision regions	convex polygons cut by bisectors. Unequal priors shift each line toward the less likely point	PS CH8.4.1
Binary error	$Q\left(\sqrt{d^2/2N_0}\right)$ for equal priors	PS CH8.3.3, 8.4.2
Union bound	$P_e \le \frac{1}{M} \sum_i \sum_{j \ne i} Q\left(\sqrt{d_{ij}^2/2N_0}\right)$. The intelligent bound keeps only faces	PS CH8.4.2
Nearest neighbours	$\bar{N}_{\min} Q\left(\sqrt{d_{\min}^2/2N_0}\right)$. Close at high SNR, but not a bound	

Correlate to get $\mathbf{r}$, pick the best metric, and bound P_e from the distances. Chapter 5 uses this receiver for every modulation.

Projects to try

Four optional projects use the chapter on simulated signals. Each gives an aim, what it practises, a few steps and what to look for.

ⓘ A CORRELATION RECEIVER FROM SAMPLES

Build the whole receiver for QPSK from sampled waveforms and measure how often it is wrong.
Practises: correlators computed as sums over samples. Scaling noise to a chosen N_0. Counting errors to estimate P_e.

1. Build two carrier basis functions over one symbol at 100 samples.
2. Send random QPSK symbols and add Gaussian noise of variance $N_0/2$ to each sample, scaled for the sample spacing.
3. Correlate, pick the nearest point and count the errors.
4. Repeat at several N_0 and plot P_e on a log axis.

Look for: the measured points follow $2Q\left(\sqrt{2E_b/N_0}\right)$ once enough errors are counted.

ⓘ MAP AGAINST ML

See how much a receiver gains by knowing the priors.
Practises: the MAP threshold for a binary link. Why the gain grows as the priors become uneven. What happens when the assumed priors are wrong.

1. Send antipodal bits with $P(s_1) = p$ for several p from 0.5 to 0.95.
2. Decide each with the midpoint and with the MAP threshold.
3. Plot both error rates against p.
4. Decide with a threshold built for the wrong p.

Look for: the two receivers agree at $p = 0.5$. The MAP receiver pulls ahead as p grows, and a wrong prior can do worse than ML.

ⓘ DECISION REGIONS ON A GRID

Draw the regions of any constellation by deciding every point of a fine grid.
Practises: minimum distance as a rule you can run anywhere. How priors move the boundaries. Why the regions are convex polygons.

1. Pick a constellation of five to eight points.
2. Decide every point of a 400×400 grid and colour it by the answer.
3. Give one point a larger prior and redraw.
4. Compare the boundaries with the perpendicular bisectors.

Look for: every boundary is a straight line. A larger prior pushes its point's boundaries outward, parallel to themselves.

TIGHTNESS OF THE UNION BOUND

Compare the union bound and its variants with a simulation across SNR.

Practises: the general and intelligent union bounds. The nearest-neighbour form. Where each is close to the truth.

1. Take 8-PSK and 16-QAM at unit average energy.
2. Compute every bound from the pairwise distances.
3. Simulate at SNR from 0 to 20 dB.
4. Plot all on one log axis and read the gap at each SNR.

Look for: at low SNR the general bound can pass 1. At high SNR all the curves close in on the simulation.

CHAPTER

5

Digital modulation methods

A carrier has an amplitude, a phase and a frequency. Keying one of them, or two at once, puts bits on the carrier. Every scheme of this chapter is a set of points. Its error rate follows from their distances, through the receiver of Chapter 4. More points carry more bits. PSK and QAM pay for them in energy, and orthogonal signals pay in band.

5.1 Putting bits on a carrier

Bits on a carrier

A baseband signal $s(t)$ has its spectrum near $f = 0$. Multiplying it by a carrier $\cos(2\pi f_c t)$ moves that spectrum to $\pm f_c$. To see this, write the cosine as two complex exponentials.

$$u(t) = s(t)\cos(2\pi f_c t) = \tfrac{1}{2}s(t)\,e^{j2\pi f_c t} + \tfrac{1}{2}s(t)\,e^{-j2\pi f_c t}$$

Multiplying a signal by $e^{j2\pi f_c t}$ shifts its spectrum up by f_c. Apply this to each term.

MODULATION SHIFTS THE SPECTRUM

$$u(t) = s(t)\cos(2\pi f_c t) \implies U(f) = \tfrac{1}{2}S(f - f_c) + \tfrac{1}{2}S(f + f_c)$$

Each copy keeps the shape of $S(f)$ at half the height.

The carrier also halves the energy. Use $\cos^2 x = \frac{1}{2} + \frac{1}{2}\cos 2x$ under the integral.

$$\int s^2(t)\cos^2(2\pi f_c t)\,dt = \tfrac{1}{2}\int s^2(t)\,dt + \tfrac{1}{2}\int s^2(t)\cos(4\pi f_c t)\,dt$$
$$\approx \tfrac{1}{2}E_s$$

The second integral nearly vanishes when $f_c \gg W$. There $s^2(t)$ changes slowly while the cosine swings through many cycles, so its positive and negative parts cancel.

Each copy runs from $f_c - W$ to $f_c + W$. The passband signal needs a band of $2W$, twice the baseband width. For example, $W = 5$ kHz on a 40 kHz carrier occupies 35 to 45 kHz, a band of $2W = 10$ kHz.

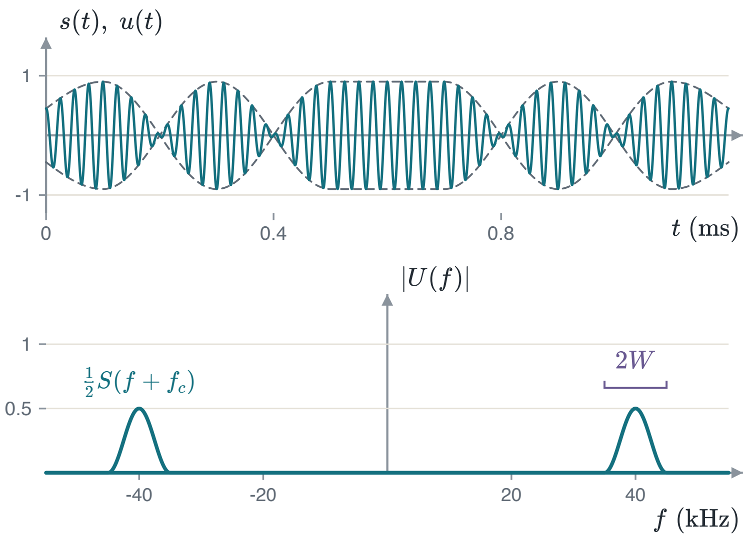


FIGURE 5.1 A baseband pulse stream $s(t)$ with band $W = 5$ kHz, and its product $u(t)$ with a 40 kHz carrier (top). The spectrum of $u(t)$ holds two half-height copies at $\pm f_c$, each $2W$ wide (bottom).

The IQ modulator

Two baseband numbers can share one carrier frequency. The IQ modulator puts I on the cosine and Q on the sine.

> THE IQ MODULATOR
>
> $$s(t) = I\cos(2\pi f_c t) - Q\sin(2\pi f_c t)$$
>
> Two baseband numbers ride two carriers 90° apart. One point (I, Q) of the plane is one symbol.

The same point can be written in polar form. Put $I = A\cos\theta$ and $Q = A\sin\theta$, then use $\cos(a+b) = \cos a\cos b - \sin a\sin b$.

$$\begin{aligned} A\cos(2\pi f_c t + \theta) &= A\cos\theta\cos(2\pi f_c t) - A\sin\theta\sin(2\pi f_c t) \\ &= I\cos(2\pi f_c t) - Q\sin(2\pi f_c t) \end{aligned}$$

> AMPLITUDE AND PHASE
>
> $$s(t) = A\cos(2\pi f_c t + \theta), \qquad A = \sqrt{I^2 + Q^2}, \quad \theta = \operatorname{atan2}(Q, I)$$
>
> PSK moves only θ and ASK moves only A.

The cosine and the sine carriers are orthogonal. Their product is $\cos x \sin x = \frac{1}{2}\sin 2x$, which averages to zero over many carrier cycles.

$$\int \cos(2\pi f_c t)\sin(2\pi f_c t)\,dt = \tfrac{1}{2}\int \sin(4\pi f_c t)\,dt \approx 0$$

So a correlator on the cosine sees I alone, and one on the sine sees Q alone. For example, the symbol $(I, Q) = (-1, +1)$ lies in the second quadrant, so its carrier phase is $\theta = \operatorname{atan2}(1, -1) = 135^\circ$.

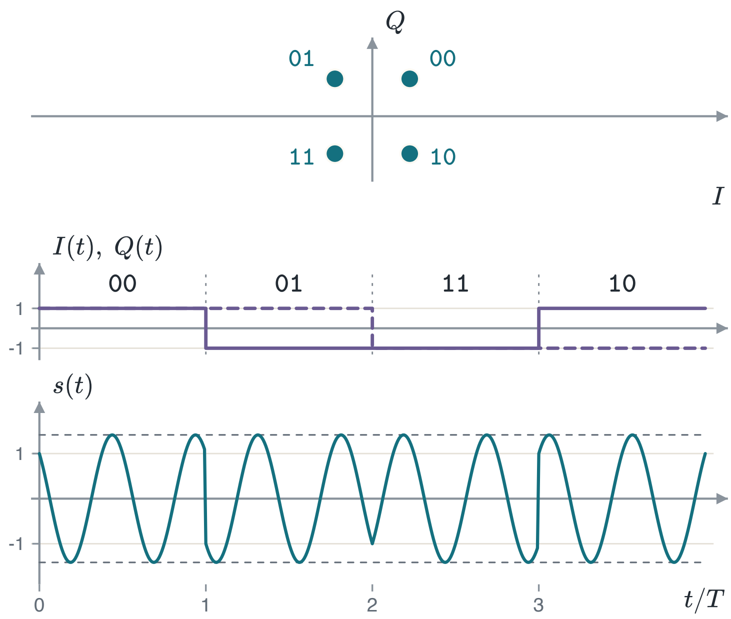


FIGURE 5.2 Four QPSK symbols at $(\pm 1, \pm 1)$, sent in the order 00, 01, 11, 10. Each point sets I (solid) and Q (dashed) for one symbol, and the carrier burst below follows them.

Binary phase-shift keying

Binary phase-shift keying (BPSK) sends a one as the carrier and a zero as the carrier inverted. The phase is 0 or π.

> BPSK
>
> $$s_1(t) = \sqrt{\frac{2E_b}{T_b}}\cos(2\pi f_c t), \qquad s_0(t) = -s_1(t)$$

Take the basis function $\psi(t) = \sqrt{2/T_b}\cos(2\pi f_c t)$. Then $s_1(t) = \sqrt{E_b}\,\psi(t)$ and $s_0(t) = -\sqrt{E_b}\,\psi(t)$. The two points sit on one axis at $\pm\sqrt{E_b}$, so they are antipodal.

$$d_{\min} = \sqrt{E_b} - (-\sqrt{E_b}) = 2\sqrt{E_b}$$

The binary result of Chapter 4 is $P_b = Q\left(\sqrt{d^2/2N_0}\right)$. Here $Q(x) = \frac{1}{2}\operatorname{erfc}(x/\sqrt{2})$ is the Gaussian tail. Substitute $d^2 = 4E_b$.

$$\frac{d^2}{2N_0} = \frac{4E_b}{2N_0} = \frac{2E_b}{N_0}$$

> KEY RESULT · BPSK
>
> $$d_{\min} = 2\sqrt{E_b}, \qquad P_b = Q\left(\sqrt{\frac{2E_b}{N_0}}\right)$$
>
> BPSK is baseband antipodal signalling times $\cos(2\pi f_c t)$. The carrier changes the band, not the error rate. With $E_b = 4$ the two points are $2\sqrt{4} = 4$ apart.

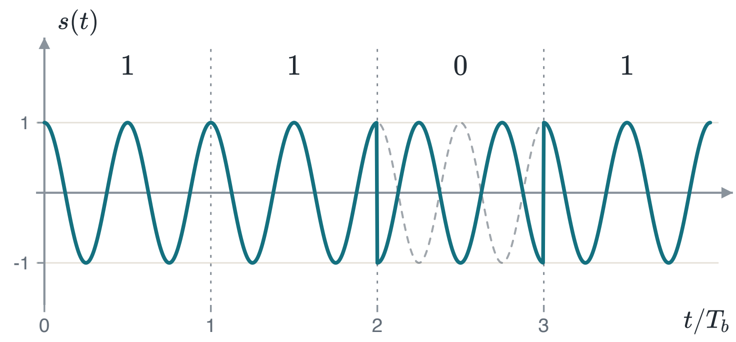


FIGURE 5.3 BPSK for the bits $1\,1\,0\,1$, two carrier cycles a bit, over the faint unmodulated carrier. A one sends the carrier and a zero sends it inverted.

Binary frequency-shift keying

Binary frequency-shift keying (BFSK) sends a one and a zero as two tones of energy E_b each.

BFSK

$$s_i(t) = \sqrt{\frac{2E_b}{T_b}}\cos(2\pi f_i t), \qquad f_1 = f_0 + \Delta f$$

The two tones are orthogonal only for certain spacings Δf. Their correlation over one bit measures how far they are from orthogonal. Expand the product with $\cos a \cos b = \frac{1}{2}\cos(a-b) + \frac{1}{2}\cos(a+b)$.

$$\rho = \frac{1}{E_b}\int_0^{T_b} s_0(t)\, s_1(t)\, dt = \frac{1}{T_b}\int_0^{T_b}\Big[\cos(2\pi\Delta f\, t) + \cos\big(2\pi(f_0+f_1)t\big)\Big]dt$$

The sum-frequency term averages out when $f_0 \gg 1/T_b$. The difference term integrates to a sine.

$$\rho \approx \frac{1}{T_b}\left[\frac{\sin(2\pi\Delta f\, t)}{2\pi\Delta f}\right]_0^{T_b} = \frac{\sin(2\pi\Delta f T_b)}{2\pi\Delta f T_b}$$

CORRELATION

$$\rho = \frac{1}{E_b}\int_0^{T_b} s_0(t)\, s_1(t)\, dt \approx \frac{\sin(2\pi\Delta f T_b)}{2\pi\Delta f T_b}$$

$\rho = 0$ at $\Delta f = n/(2T_b)$. The first zero comes at $2\pi\Delta f T_b = \pi$, so $\Delta f = 1/(2T_b)$ is the least spacing that keeps the tones orthogonal. The lowest value, $\rho = -0.217$, falls at $\Delta f = 0.715/T_b$.

At an orthogonal spacing the two points lie on two axes at right angles, each $\sqrt{E_b}$ from the origin. By Pythagoras, $d^2 = E_b + E_b = 2E_b$.

KEY RESULT · BFSK

$$d_{\min} = \sqrt{2E_b}, \qquad P_b = Q\left(\sqrt{\frac{E_b}{N_0}}\right)$$

Two orthogonal points have $d^2 = 2E_b$, against $4E_b$ for BPSK. BFSK needs 3 dB more energy.

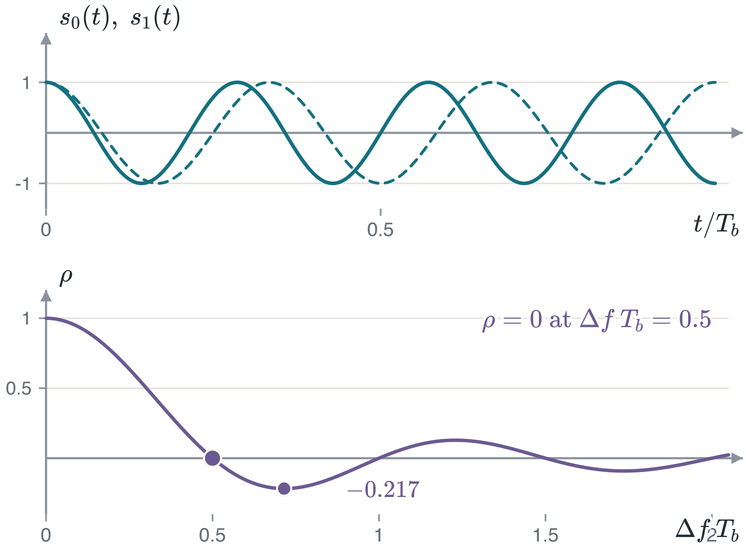


FIGURE 5.4 Two tones over one bit, $f_0 T_b = 3$ (dashed) and $f_0 + \Delta f$ (solid), at $\Delta f\, T_b = 0.5$ (top). Their correlation against the spacing: $\rho = 0$ first at $\Delta f = 1/(2T_b)$, and $\rho = -0.217$ is the lowest value (bottom).

Binary amplitude-shift keying

Binary amplitude-shift keying (BASK), also called on-off keying, sends the carrier for a one and nothing for a zero.

BASK

$$s_1(t) = \sqrt{\frac{2E}{T_b}} \cos(2\pi f_c t), \qquad s_0(t) = 0$$

Ones and zeros are equally likely, and only the ones cost energy. The average energy a bit is therefore half the peak energy E.

$$E_b = \tfrac{1}{2}E + \tfrac{1}{2}(0) = \frac{E}{2}$$

The two points sit at 0 and $\sqrt{E}$ on one axis. Write their distance in terms of E_b.

$$d = \sqrt{E} - 0 = \sqrt{E} = \sqrt{2E_b}$$

KEY RESULT · BASK

$$d = \sqrt{E} = \sqrt{2E_b}, \qquad P_b = Q\left(\sqrt{\frac{E_b}{N_0}}\right)$$

This is the distance of BFSK at the same average energy. For example, a peak energy $E = 2$ gives $E_b = E/2 = 1$.

⊗ COMMON ERROR

Use the average $E_b = E/2$, not the peak E, in P_b. Only the ones carry energy.

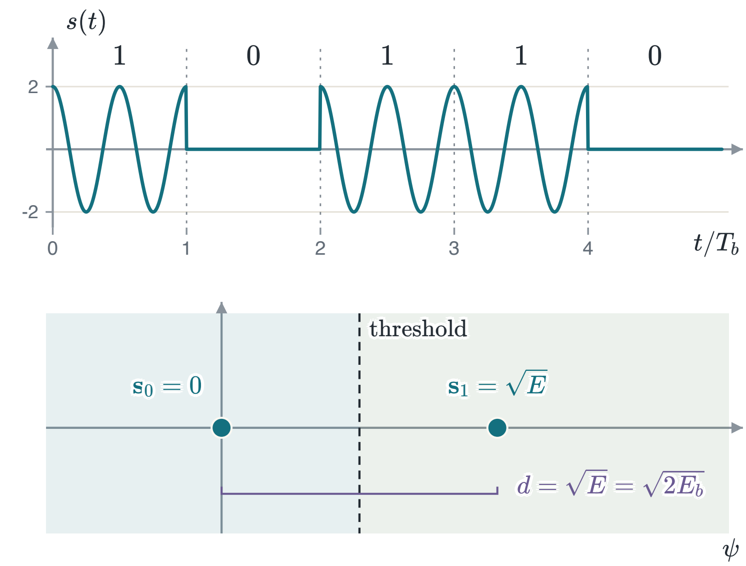


FIGURE 5.5 On-off keying of $1\,0\,1\,1\,0$ with peak energy $E = 2$ a bit (top). Its two points on one axis, with the threshold halfway (bottom).

Three binary schemes compared

The three schemes differ only in the squared distance between their two points.

SQUARED DISTANCE

$$\text{BPSK: } d^2 = 4E_b, \qquad \text{BFSK, BASK: } d^2 = 2E_b$$

Half the squared distance costs a factor 2 in energy, $10\log_{10} 2 = 3$ dB.

To give the same argument of Q, BFSK needs E_b/N_0 twice as large as BPSK does. So BPSK reaches $P_b = 10^{-5}$ at 9.6 dB, and BFSK reaches it at $9.6 + 3.0 = 12.6$ dB.

BFSK and BASK need the same average E_b/N_0. BASK spends it only on the ones.

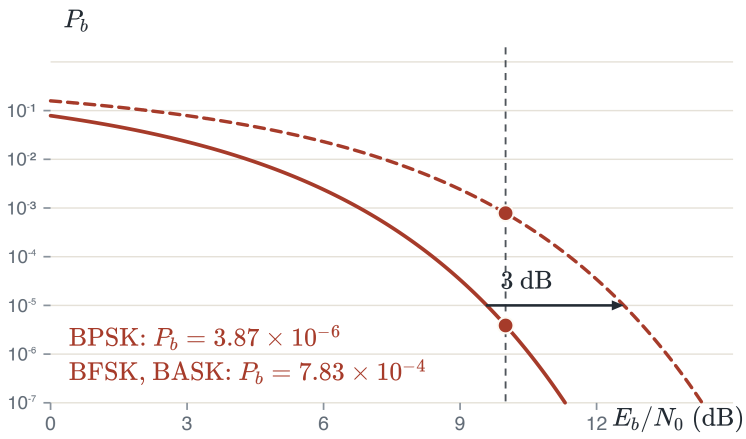


FIGURE 5.6 Bit error of BPSK (solid) and of BFSK and BASK (dashed). At 10^{-5} the dashed curve sits 3 dB to the right. The dots read both curves at $E_b/N_0 = 10$ dB, the case of Example 5.1.

EXAMPLE 5.1 — A BINARY LINK

GIVEN A link receives $P = 4 \times 10^{-15}$ W at $R_b = 100$ kb/s, with $N_0 = 4 \times 10^{-21}$ W/Hz.

FIND P_b for BPSK and for BFSK.

METHOD Both error formulas depend only on E_b/N_0. The energy a bit is power times bit time, $E_b = P/R_b$. Divide it by N_0.

SOLUTION $E_b = P/R_b = 4 \times 10^{-15}/10^5 = 4 \times 10^{-20}$ J, so $E_b/N_0 = 10$, or 10 dB. BPSK: $P_b = Q(\sqrt{20}) = 3.87 \times 10^{-6}$. BFSK: $P_b = Q(\sqrt{10}) = 7.83 \times 10^{-4}$.

CHECK BFSK needs twice the energy for the same distance. It would match BPSK at 8×10^{-15} W.

COMMON ERROR

Use $E_b = P/R_b$, not P, over N_0. The ratio $P/N_0 = 10^6$ Hz is not a signal-to-noise ratio.

Binary keying around us

- An NFC reader sends 106 kb/s by switching the amplitude of a 13.56 MHz carrier, $s(t) = A(t)\cos(2\pi f_c t)$. One bit lasts 9.43 μs.
- FM radio sends RDS data at 1187.5 b/s by phase keying a 57 kHz subcarrier, $s(t) = \pm\cos(2\pi f_c t)$. One bit holds 48 carrier cycles.
- A car key fob switches a 433.92 MHz carrier on and off. Bit n sends $A[n]\cos(2\pi f_c t)$ with $A[n] \in \{0, 1\}$.
- GPS phase-keys a 1575.42 MHz carrier at 1.023 Mchip/s. Its spectrum $T_c \operatorname{sinc}^2((f - f_c)T_c)$, with $\operatorname{sinc} x = \sin(\pi x)/(\pi x)$, has nulls at ± 1.023 MHz.

- **One switch a bit.** Each system changes one thing about its carrier for each bit: the amplitude or the sign.
- **Energy on the ones.** On-off keying spends energy only on the ones. With equal ones and zeros, E_b is half the peak energy.
- **Bit rate and band.** A faster bit rate widens the spectrum. The GPS main lobe is $2 \times 1.023 = 2.046$ MHz wide.

5.2 Phase-shift keying

M-ary phase-shift keying

M-ary PSK puts M points on a circle of radius $\sqrt{E_s}$. All points have equal energy and differ only in phase. Each symbol carries $k = \log_2 M$ bits, so $E_s = kE_b$.

M-PSK

$$s_m(t) = \sqrt{\frac{2E_s}{T}} \cos\left(2\pi f_c t + \frac{2\pi m}{M}\right), \quad m = 0, \ldots, M-1$$

Two neighbours are each $\sqrt{E_s}$ from the origin, and the angle between them is $2\pi/M$. The cosine rule gives their squared distance.

$$\begin{aligned} d_{\min}^2 &= E_s + E_s - 2E_s \cos\frac{2\pi}{M} \\ &= 2E_s\left(1 - \cos\frac{2\pi}{M}\right) \\ &= 4E_s \sin^2\frac{\pi}{M} \end{aligned}$$

The last line uses $1 - \cos 2x = 2\sin^2 x$ with $x = \pi/M$.

NEIGHBOUR DISTANCE

$$d_{\min} = 2\sqrt{E_s}\,\sin\frac{\pi}{M}$$

At $M = 2$, $d_{\min} = 2\sqrt{E_s}$: this is BPSK. Each doubling of M nearly halves $d_{\min}$. QPSK with $E_s = 2$ has $d_{\min} = 2\sqrt{2}\,\sin(\pi/4) = 2\sqrt{2}/\sqrt{2} = 2$.

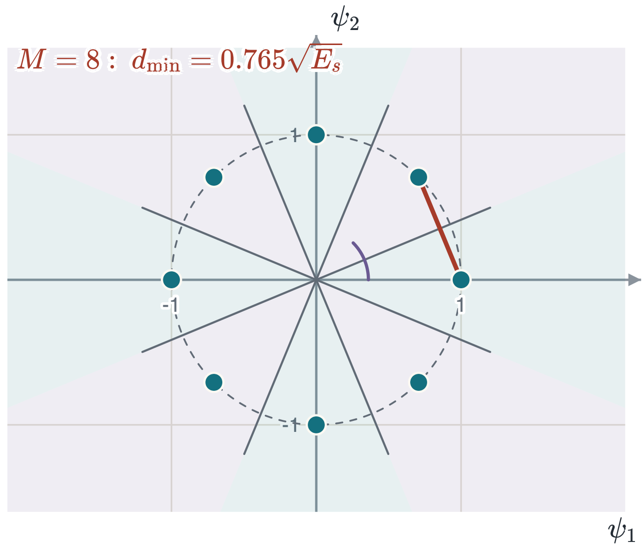


FIGURE 5.7 8-PSK with $E_s = 1$ and its wedges. The red chord is $d_{\min} = 2\sin(\pi/8)\sqrt{E_s} = 0.765\sqrt{E_s}$.

QPSK as two BPSK links

QPSK is M-PSK with $M = 4$. Written through the IQ modulator, it splits into a cosine half and a sine half. Each half is a BPSK signal.

$$s(t) = \underbrace{I\cos(2\pi f_c t)}_{\text{BPSK on cos}} - \underbrace{Q\sin(2\pi f_c t)}_{\text{BPSK on sin}}$$

The carriers are orthogonal, so the noise on one half does not reach the other. The symbol energy $E_s = 2E_b$ splits equally, and each bit sees its own BPSK link with energy E_b.

KEY RESULT · QPSK

$$P_b = Q\left(\sqrt{\frac{2E_b}{N_0}}\right), \qquad P_e = 1 - (1 - P_b)^2 \approx 2P_b$$

A symbol is right only when both halves are right. QPSK carries twice the bits of BPSK in the same band, at the same bit error.

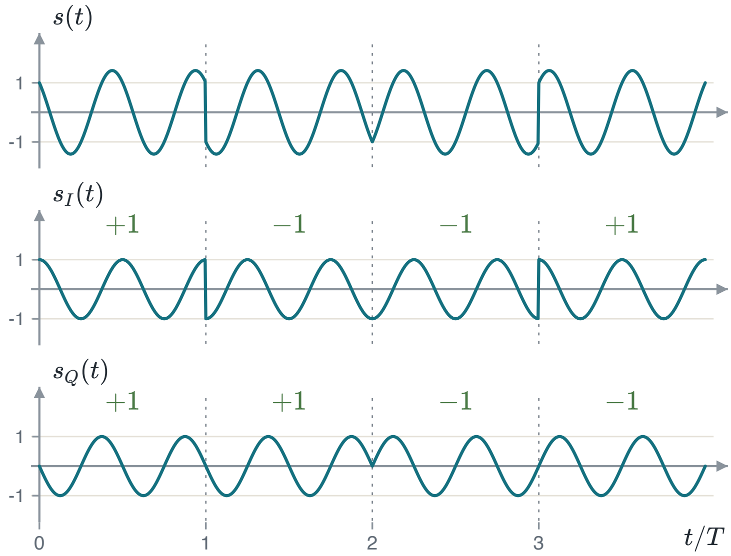


FIGURE 5.8 The QPSK burst of the IQ modulator (top) and its two halves. Each half is a BPSK signal, and the receiver decides each sign on its own (green).

Detecting PSK by phase

The receiver uses two correlators on the basis of the IQ modulator.

> TWO CORRELATORS
>
> $$y_1 = \int_0^T r(t)\,\psi_1(t)\,dt, \qquad y_2 = \int_0^T r(t)\,\psi_2(t)\,dt$$
>
> $$\psi_1(t) = \sqrt{2/T}\cos(2\pi f_c t), \qquad \psi_2(t) = -\sqrt{2/T}\sin(2\pi f_c t)$$

Write the received vector $\mathbf{y}$ in polar form, with length $\|\mathbf{y}\|$ and angle θ. Its squared distance to the point $\mathbf{s}_m$ follows from the cosine rule.

$$\|\mathbf{y} - \mathbf{s}_m\|^2 = \|\mathbf{y}\|^2 + E_s - 2\|\mathbf{y}\|\sqrt{E_s}\,\cos\Big(\theta - \frac{2\pi m}{M}\Big)$$

Only the cosine depends on m. The distance is least where that cosine is largest, which is where $2\pi m/M$ is nearest θ.

> PHASE DECISION
>
> $$\theta = \operatorname{atan2}(y_2, y_1), \qquad \hat{m} = \text{the } m \text{ with } \tfrac{2\pi m}{M} \text{ nearest } \theta$$
>
> All points have equal energy, so the nearest point has the nearest phase. The length of $\mathbf{y}$ plays no part.

For example, an 8-PSK receiver measures $\theta = 50^\circ$. The phases are 45° apart, and 50° is 5° from 45° and 40° from 90°. The receiver picks 45°.

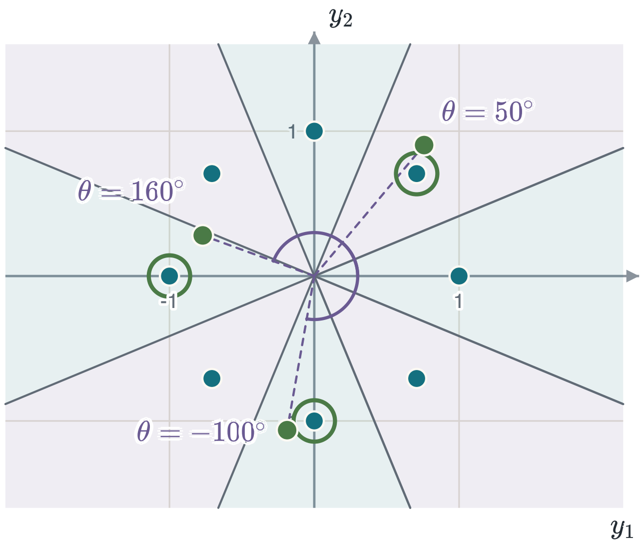


FIGURE 5.9 8-PSK and its wedges, with received points at $\theta = 50°$, $160°$ and $-100°$. For each one the receiver picks the point of nearest phase (green ring).

M-PSK error probability

Each point has two neighbours at $d_{\min}$. The nearest-neighbour form of Chapter 4 is $P_e \approx \bar{N}_{\min}\, Q\left(\sqrt{d_{\min}^2/2N_0}\right)$, with $\bar{N}_{\min} = 2$. Substitute the neighbour distance.

$$\frac{d_{\min}^2}{2N_0} = \frac{4E_s \sin^2(\pi/M)}{2N_0} = \frac{2E_s}{N_0} \sin^2 \frac{\pi}{M}$$

SYMBOL ERROR

$$P_e \approx 2Q\left(\sqrt{\frac{2E_s}{N_0}} \sin \frac{\pi}{M}\right), \qquad E_s = kE_b$$

Doubling M brings the points closer. To keep the same $d_{\min}$, the symbol energy must grow by the ratio of the two $\sin^2$ terms. From QPSK to 8-PSK:

$$\frac{E_{s,8}}{E_{s,4}} = \frac{\sin^2(\pi/4)}{\sin^2(\pi/8)} = \frac{0.5}{0.1464} = 3.41\ (5.33\ \text{dB})$$

Each 8-PSK symbol now carries 3 bits, not 2. Per bit the cost is $5.33 - 10\log_{10} 1.5 = 5.33 - 1.76 = 3.57$ dB. For large M, $\sin(\pi/M) \approx \pi/M$, so each doubling divides $\sin^2$ by 4. The cost then tends to 6 dB a doubling.

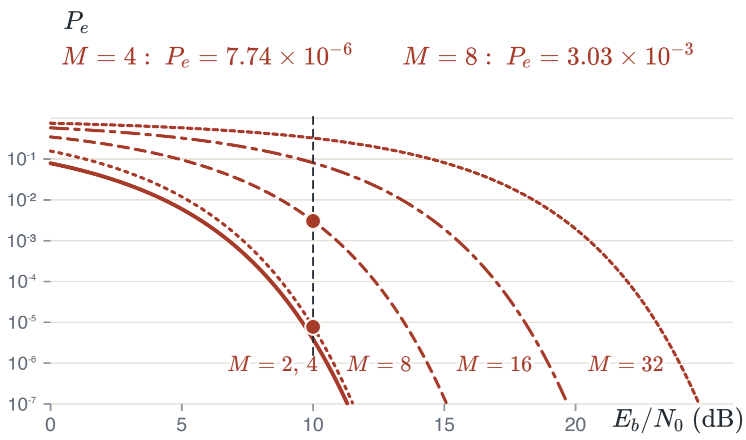


FIGURE 5.10 Symbol error of M-PSK for $M = 2$ to 32, from the nearest-neighbour form, exact for $M = 2$. $M = 2$ and 4 nearly share one curve. The dots read QPSK and 8-PSK at $E_b/N_0 = 10$ dB.

Gray labels

A symbol error nearly always lands on a neighbour. Gray labels make neighbours differ in one bit.

GRAY CODE

$$g = i \oplus \lfloor i/2 \rfloor$$

For $i = 0, \ldots, 7$ this gives 000, 001, 011, 010, 110, 111, 101, 100. Neighbours differ in one bit.

For example, $i = 5$ is 101 and $\lfloor 5/2 \rfloor = 2$ is 010. Their bitwise sum is 101 $\oplus$ 010 $=$ 111, the sixth label of the list.

With Gray labels a neighbour costs one bit of the k bits of a symbol. The bit error is then the symbol error divided by k.

BITS FROM SYMBOLS

$$P_b \approx \frac{P_e}{\log_2 M}$$

For example, 8-PSK with Gray labels at $P_e = 3 \times 10^{-3}$ has $P_b \approx P_e/3 = 10^{-3}$.

⚠ NATURAL LABELS

Natural labels put 011 beside 100. That one symbol error costs three bits.

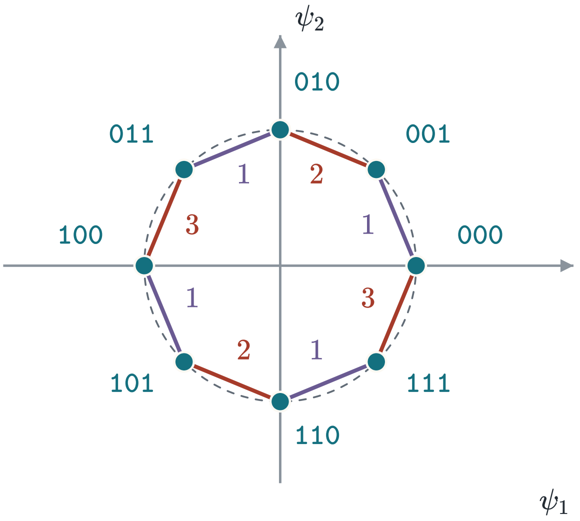


FIGURE 5.11 8-PSK with natural labels. Each chord shows how many bits its two labels differ in. The chords between 011 and 100 and between 111 and 000 cost three.

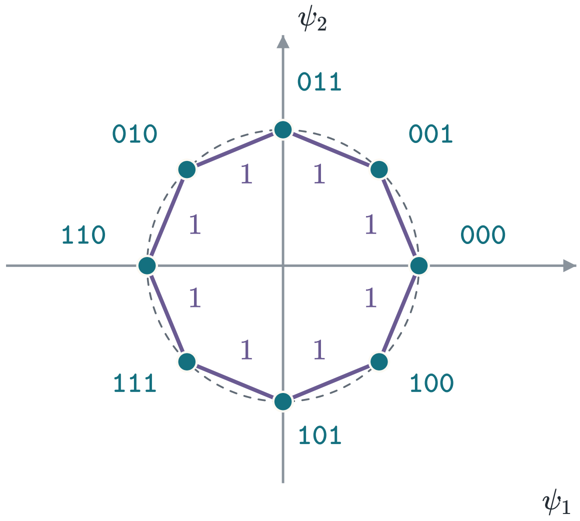


FIGURE 5.12 8-PSK with Gray labels. Every chord costs one bit.

EXAMPLE 5.2 – 8-PSK

GIVEN 8-PSK with Gray labels at $E_b/N_0 = 10$ dB.

FIND The symbol error P_e and the bit error P_b.

METHOD Use the nearest-neighbour form, since each point has two neighbours at d_{min}. First turn E_b/N_0 into E_s/N_0. Then divide P_e by 3 for Gray labels.

SOLUTION 10 dB is a ratio of 10, and each symbol carries 3 bits, so $E_s/N_0 = 3(10) = 30$. The argument of Q is $x = \sqrt{2(30)}\,\sin(\pi/8) = 2.96$. So $P_e \approx 2Q(2.96) = 2(1.52 \times 10^{-3}) = 3.0 \times 10^{-3}$ and $P_b \approx P_e/3 = 1.0 \times 10^{-3}$.

CHECK The sent point sits inside a wedge of $\pm 22.5^\circ$. Each of its two neighbours, d_{min} away, contributes one Q term.

COMMON ERROR

Use $E_s = 3E_b$, not E_b, inside Q. With E_b the argument is 1.71 and P_e comes out as 0.087.

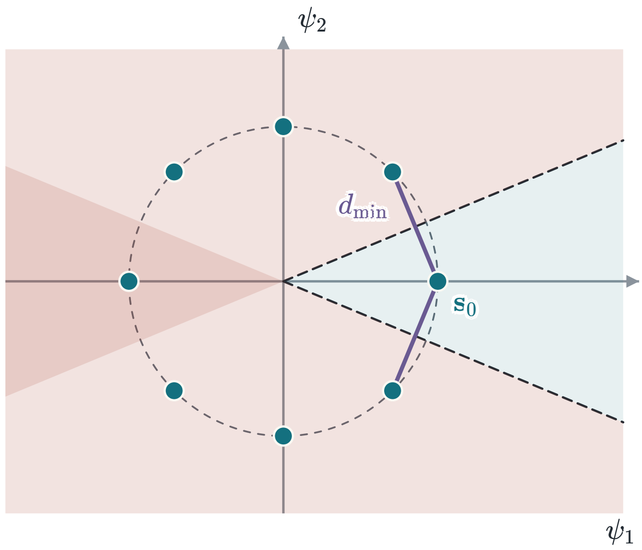


FIGURE 5.13 Example 5.2: the sent point $\mathbf{s}_0$ and its wedge of $\pm 22.5°$. The red half-planes lie nearer a neighbour, each $d_{\min}$ away.

A carrier phase error

A receiver whose carrier is off by φ receives

A PHASE ERROR

$$r(t) = \sqrt{\frac{2E_s}{T}} \cos(2\pi f_c t + \theta_m + \varphi) + n(t)$$

It sees every point turned by φ.

⚠ AMBIGUITY

M-PSK looks the same after a turn of $2\pi/M$. A receiver can lock its carrier to any of these M phases.

For example, take QPSK with $\varphi = 90°$ and no noise. Each point lands exactly on a neighbour, so every symbol is decided as that neighbour. No decision is right.

Differential encoding avoids the problem. It sends the bits in the change of phase from one symbol to the next. A constant φ then cancels.

FIGURE 5.14 QPSK with 200 noisy symbols at a carrier phase error $\varphi = 20^\circ$, decided in the fixed quadrants. The green rings are the points turned by φ. Red dots are wrong decisions.

Differential PSK

Differential PSK (DPSK) puts each bit in the change of phase, not in the phase itself.

DIFFERENTIAL ENCODING

$$\theta_n = \theta_{n-1} + \pi b_n \pmod{2\pi}$$

For example, send the bits $1\,0\,1\,1$ from $\theta_0 = 0$. A one adds π and a zero adds nothing.

$$\theta_1 = \pi, \quad \theta_2 = \pi, \quad \theta_3 = 2\pi \equiv 0, \quad \theta_4 = \pi$$

Three ones add 3π, which is π modulo 2π. So the final phase is $\theta_4 = \pi$.

The receiver compares each received point with the one before. Their dot product is $\|\mathbf{y}_n\|\|\mathbf{y}_{n-1}\|\cos(\theta_n - \theta_{n-1})$, which is negative when the phase has changed by π.

DIFFERENTIAL DETECTION

$$\hat{b}_n = 1 \quad \text{when} \quad \mathbf{y}_n \cdot \mathbf{y}_{n-1} < 0$$

A constant phase error turns both points by the same angle. A turn does not change the dot product, so the error cancels.

⚠ ERRORS IN PAIRS

One badly received symbol spoils two comparisons, with the symbol before and the one after. DPSK errors tend to come in pairs.

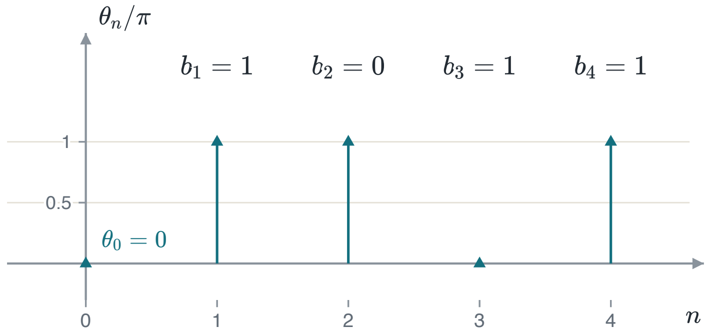


FIGURE 5.15 Differential encoding of the bits $1\,0\,1\,1$ from $\theta_0 = 0$. A one adds π to the phase and a zero adds nothing, so θ_n/π runs $0, 1, 1, 0, 1$.

DPSK error probability

Binary DPSK needs no carrier phase. The previous symbol is the reference.

> KEY RESULT · BINARY DPSK
>
> $$P_b = \frac{1}{2} e^{-E_b/N_0}$$

The price is small. Solve $\frac{1}{2}e^{-x} = 10^{-5}$ for the ratio $x = E_b/N_0$.

$$x = \ln\left(5 \times 10^4\right) = 10.82, \qquad 10\log_{10} 10.82 = 10.34\ \text{dB}$$

At $P_b = 10^{-5}$ binary DPSK needs 10.34 dB against 9.59 dB for BPSK. It costs 0.75 dB, less than 1 dB. At $E_b/N_0 = 10$ dB, $P_b = \frac{1}{2}e^{-10} = 2.27 \times 10^{-5}$, where BPSK gives 3.87×10^{-6}.

> ⚠ FOUR PHASES AND MORE
>
> For $M \geq 4$, differential detection costs about 3 dB over coherent PSK. The reference symbol is as noisy as the one decided.

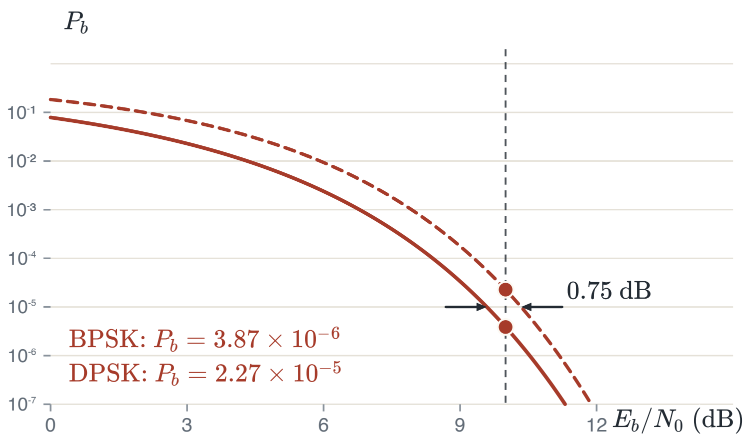


FIGURE 5.16 Bit error of coherent BPSK (solid) and binary DPSK (dashed). At 10^{-5} they are 0.75 dB apart. The dots read both at $E_b/N_0 = 10$ dB.

Phase keying around us

- 802.11b Wi-Fi at 1 Mb/s sends one bit a microsecond as a phase change of 0 or π: $\theta_n = \theta_{n-1} + \pi b_n$. At 2 Mb/s it uses four phase changes.
- Satellite television (DVB-S) sends QPSK with Gray labels. Each bit sees its own BPSK link, $P_b = Q\big(\sqrt{2E_b/N_0}\big)$.
- Bluetooth EDR at 2 Mb/s uses $\pi/4$-DQPSK. Each symbol turns the phase by $\Delta\theta \in \{\pm\pi/4, \pm 3\pi/4\}$, so no step passes through the origin.
- Satellite DVB-S2 adds 8PSK to QPSK. Neighbouring points are $d_{\min} = 2\sqrt{E_s}\sin(\pi/8) = 0.765\sqrt{E_s}$ apart.

- **Phase alone.** PSK keeps the amplitude constant. The power amplifier sees one level.
- **Differential phase.** 802.11b and Bluetooth EDR carry the bits in phase changes. The receiver needs no absolute carrier phase.
- **Crowding.** Beyond eight points the circle crowds. Satellite links then add a second ring, and other links move to QAM.

5.3 Amplitude and quadrature

M-ary amplitude-shift keying

M-ary amplitude-shift keying (M-ASK), also called PAM, puts M equally spaced amplitudes on one carrier. Neighbours are d apart.

> M-ASK
>
> $$s_m = (2m - 1 - M)\,\frac{d}{2}, \qquad m = 1, \ldots, M$$
>
> The amplitudes are $\pm d/2$, $\pm 3d/2$, and so on.

The average energy is the mean of the squared amplitudes, with each level equally likely.

$$E_s = \frac{1}{M}\sum_{m=1}^{M} s_m^2 = \frac{d^2}{4M}\sum_{m=1}^{M}(2m-1-M)^2$$

The numbers $2m - 1 - M$ run over the odd values $\pm 1, \pm 3, \ldots, \pm(M-1)$. Their squares sum to $M(M^2-1)/3$. For $M = 4$ that is $1 + 9 + 1 + 9 = 20 = 4(15)/3$.

> AVERAGE ENERGY
>
> $$E_s = \frac{d^2}{4M}\cdot\frac{M(M^2-1)}{3} = \frac{(M^2-1)\,d^2}{12}$$
>
> 4-ASK with levels ± 1, ± 3 has $d = 2$ and $E_s = (1+9+9+1)/4 = 5$, or $(16-1)(4)/12 = 5$.

Count the neighbours. The average count is $\bar{N}_{\min} = \big(2(M-2)+2\big)/M = 2(M-1)/M$. Then write d in terms of E_s.

$$d^2 = \frac{12E_s}{M^2-1} \quad \Longrightarrow \quad \frac{d^2}{2N_0} = \frac{6E_s}{(M^2-1)N_0}$$

KEY RESULT · M-ASK

$$P_e = \frac{2(M-1)}{M} Q\left(\sqrt{\frac{6E_s}{(M^2-1)N_0}}\right)$$

Inner points have two neighbours and the end points one. Each doubling of M costs about 6 dB.

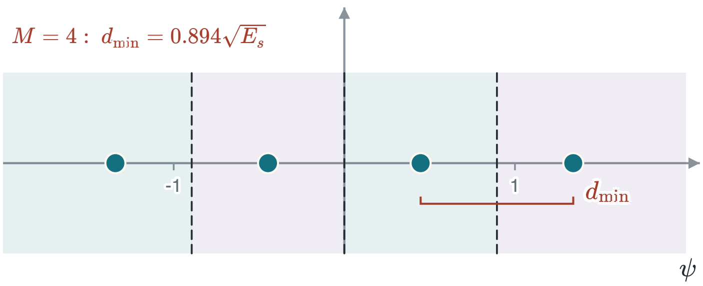


FIGURE 5.17 4-ASK on one axis at unit average energy, with its thresholds halfway between levels. The spacing shrinks as $\sqrt{12/(M^2-1)}$.

EXAMPLE 5.3 – FOUR-LEVEL ASK

GIVEN 4-ASK with levels $\pm A$, $\pm 3A$ at $E_s/N_0 = 25$.

FIND The symbol error P_e, and the average neighbour count $\bar{N}_{\min}$.

METHOD Use the nearest-neighbour form. Write $d_{\min} = 2A$ in terms of E_s, and count the neighbours of each point.

SOLUTION $E_s = A^2(9+1+1+9)/4 = 5A^2$ and $d_{\min} = 2A$, so $d^2_{\min}/2N_0 = 4A^2/2N_0 = 2E_s/5N_0 = 2(25)/5 = 10$. The end points have 1 neighbour and the inner two have 2, so $\bar{N}_{\min} = (1+2+2+1)/4 = 1.5$. Then $P_e \approx 1.5\,Q(\sqrt{10}) = 1.5(7.83 \times 10^{-4}) = 1.17 \times 10^{-3}$.

CHECK The M-ASK formula with $M = 4$ gives the same: $2(3)/4 = 1.5$ and $6(25)/15 = 10$.

COMMON ERROR

Use $\bar{N}_{\min} = 1.5$, not 2. The two end points have one neighbour each.

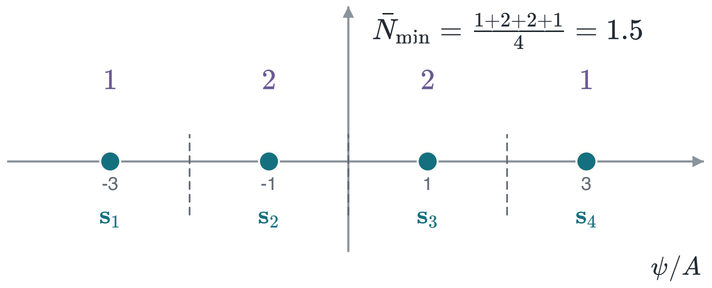


FIGURE 5.18 Example 5.3: four levels at $\pm A$, $\pm 3A$ with the neighbour count of each point, 1, 2, 2, 1, and their average 1.5.

Quadrature amplitude modulation

Quadrature amplitude modulation (QAM) runs one ASK signal on the cosine and another on the sine. Square M-QAM is two $\sqrt{M}$-level ASK axes at right angles.

SQUARE QAM

$$s(t) = I\cos(2\pi f_c t) - Q\sin(2\pi f_c t), \qquad I,\, Q \in \left\{\pm\tfrac{d}{2}, \pm\tfrac{3d}{2}, \ldots\right\}$$

Each axis carries $\sqrt{M}$ levels, d apart. Gray labels on each axis give neighbours that differ in one bit.

The energy of a point is the sum of the energies on the two axes. Each axis is a $\sqrt{M}$-level ASK, so put $\sqrt{M}$ in place of M in the ASK energy.

AVERAGE ENERGY

$$E_s = 2 \cdot \frac{\big((\sqrt{M})^2 - 1\big)\, d^2}{12} = \frac{(M-1)\, d^2}{6}$$

16-QAM with $d = 2$ gives $E_s = 15(4)/6 = 10$.

Count the neighbours of 16-QAM. Its 4 corners have 2 neighbours, its 8 edge points 3 and its 4 inner points 4.

$$\bar{N}_{\min} = \frac{4(2) + 8(3) + 4(4)}{16} = \frac{48}{16} = 3 = 4\Big(1 - \frac{1}{\sqrt{16}}\Big)$$

In general $\bar{N}_{\min} = 4(1 - 1/\sqrt{M})$. The distance in terms of E_s is $d^2 = 6E_s/(M-1)$, so $d^2/2N_0 = 3E_s/((M-1)N_0)$.

KEY RESULT · SQUARE QAM

$$P_e \approx 4\left(1 - \frac{1}{\sqrt{M}}\right) Q\left(\sqrt{\frac{3E_s}{(M-1)N_0}}\right)$$

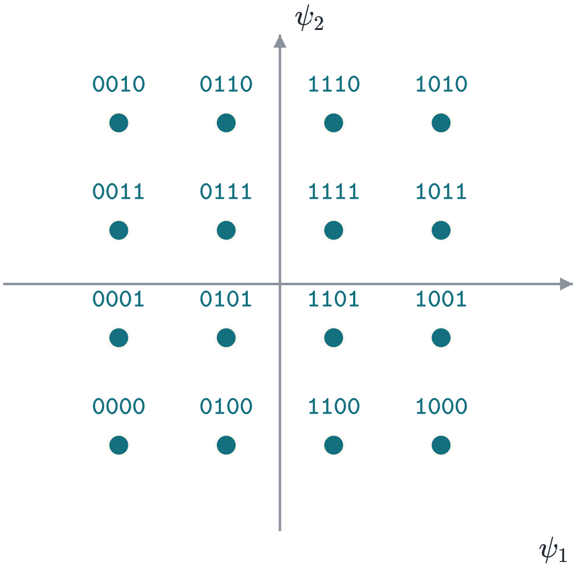


FIGURE 5.19 16-QAM as 4-ASK on the cosine and 4-ASK on the sine, with Gray labels on each axis.

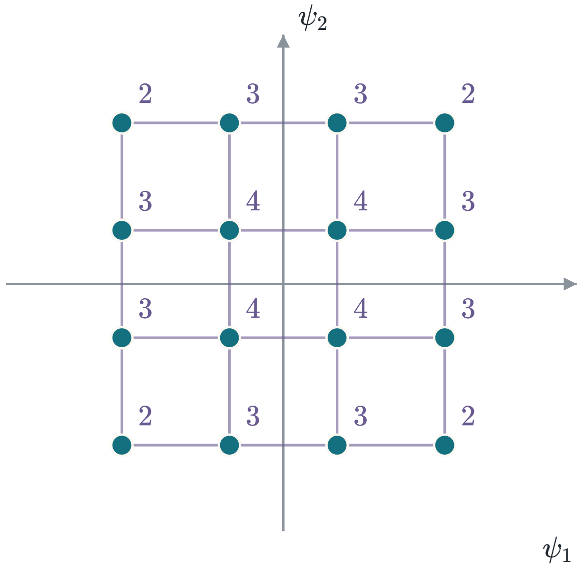


FIGURE 5.20 The neighbours of each 16-QAM point: 2 at a corner, 3 on an edge, 4 inside. Their average is $\bar{N}_{\min} = 48/16 = 3$.

QAM error probability

A symbol is right only when both axes are right. Each axis is a $\sqrt{M}$-level ASK with its own error $P_{\sqrt{M}}$.

EXACT SQUARE QAM

$$P_e = 1 - \left(1 - P_{\sqrt{M}}\right)^2, \qquad P_{\sqrt{M}} = 2\left(1 - \tfrac{1}{\sqrt{M}}\right) Q\left(\sqrt{\tfrac{3E_s}{(M-1)N_0}}\right)$$

Expand the square to compare this with the nearest-neighbour form.

$$1 - \left(1 - P_{\sqrt{M}}\right)^2 = 2P_{\sqrt{M}} - P_{\sqrt{M}}^2$$

The nearest-neighbour form keeps only $2P_{\sqrt{M}}$. At high SNR the square is tiny, so the two forms are close. For example, $P_{\sqrt{M}} = 10^{-3}$ gives $P_e = 1 - (1 - 10^{-3})^2 = 1.999 \times 10^{-3}$, about 2×10^{-3}.

⚠ THREE DB A BIT

At a fixed $d_{\min}$, E_s grows as $M - 1$. Doubling M nearly doubles $M - 1$, so each extra bit a symbol costs about 3 dB of E_s/N_0.

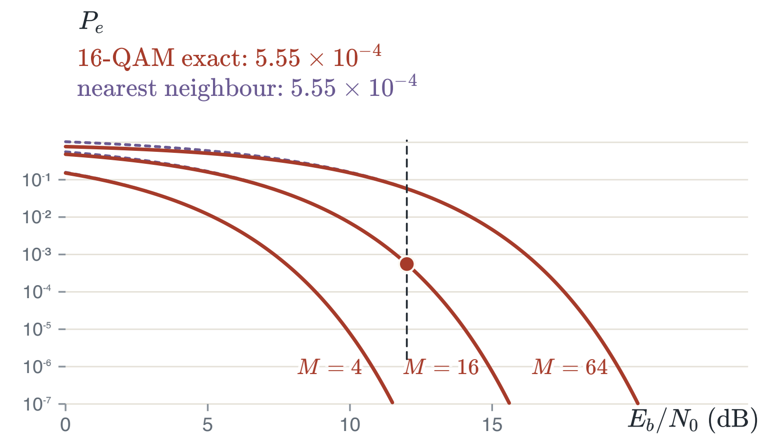


FIGURE 5.21 Symbol error of square QAM for $M = 4$, 16 and 64. Solid: exact. Dashed: nearest-neighbour form. The dot reads 16-QAM at $E_b/N_0 = 12$ dB.

Eight-point constellations

Take four sets of eight points, all with neighbours $d_{\min} = 2$ apart. At high SNR they err at about the same rate. The best set needs the least energy.

The average energy of each set is the mean of $x^2 + y^2$ over its points. Each set has four points at one radius and four at another.

$$\begin{aligned}
&\text{(a)} \quad E = \tfrac{1}{8}\big(4 \cdot 2 + 4 \cdot 10\big) = 6 \\
&\text{(b)} \quad E = \tfrac{1}{8}\big(4 \cdot 2 + 4 \cdot 2(1 + \sqrt{2})^2\big) = 6.83 \\
&\text{(c)} \quad E = \tfrac{1}{8}\big(4 \cdot 4 + 4 \cdot 8\big) = 6 \\
&\text{(d)} \quad E = \tfrac{1}{8}\big(4 \cdot 2 + 4(1 + \sqrt{3})^2\big) = 4.73
\end{aligned}$$

Set (d) puts four points on an inner square at $(\pm 1, \pm 1)$ and four on the axes at $1 + \sqrt{3}$. The point $(1 + \sqrt{3}, 0)$ is $\sqrt{3 + 1} = 2$ from $(1, 1)$, so $d_{\min}$ stays 2.

GAIN

$$10 \log_{10} \frac{6}{4.73} = 1.03 \text{ dB}$$

Set (d) saves about 1 dB over the rectangle (a). The packing matters, not the grid.

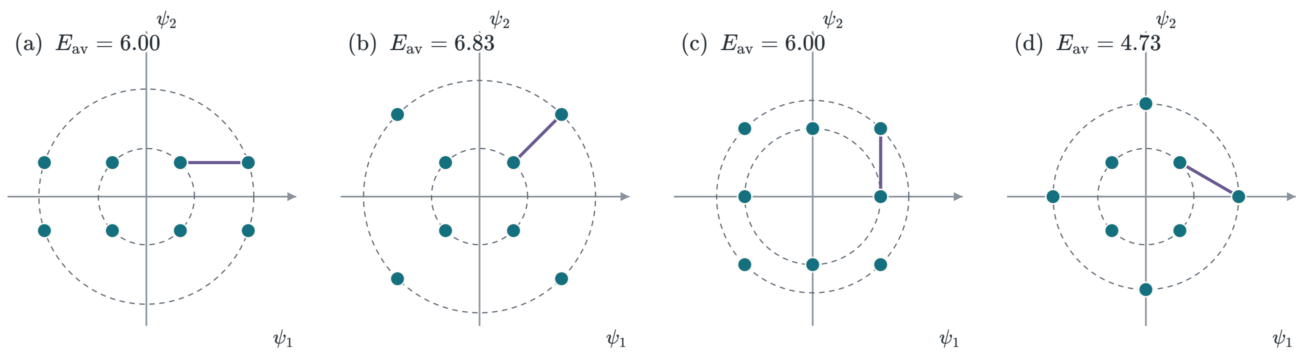


FIGURE 5.22 Four sets of eight points, each with $d_{\min} = 2$ (violet chord), and the average energy of each. Set (d) needs the least.

EXAMPLE 5.4 – 16-QAM

GIVEN 16-QAM with $d_{\min} = 2$ and $E_s/N_0 = 80$, about 19 dB.

FIND The energy E_s, the symbol error P_e and the bit error P_b with Gray labels.

METHOD Use the nearest-neighbour form of square QAM. Find E_s from d, write $d_{\min}^2/2N_0$ in terms of E_s/N_0, and divide P_e by 4 bits.

SOLUTION $E_s = (M-1)d^2/6 = 15(4)/6 = 10$. Then $d_{\min}^2/2N_0 = 3E_s/15N_0 = E_s/5N_0 = 16$, and $\bar{N}_{\min} = 3$. So $P_e \approx 3\,Q(\sqrt{16}) = 3\,Q(4) = 9.5 \times 10^{-5}$ and $P_b \approx P_e/4 = 2.4 \times 10^{-5}$.

CHECK The grid $\pm 1, \pm 3$ on both axes gives $E_s = 5 + 5 = 10$ directly. The boundaries sit halfway between levels, at 0 and ± 2 on each axis.

COMMON ERROR

Use $\bar{N}_{\min} = 3$, not 4. Only the four inner points have four neighbours.

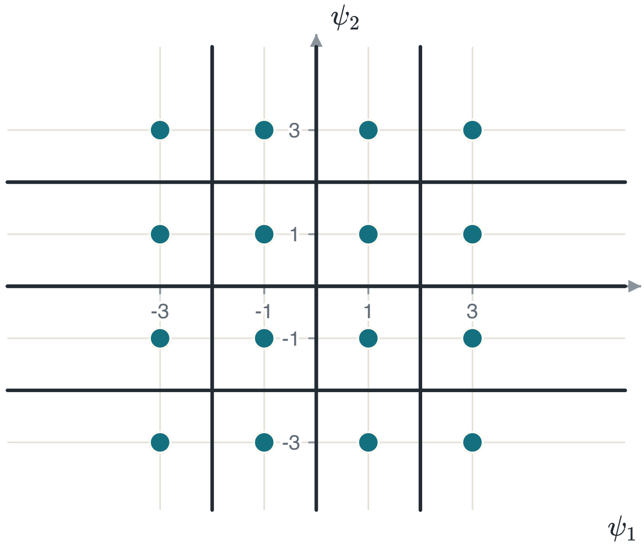


FIGURE 5.23 Example 5.4: 16-QAM on the grid ± 1, ± 3 with its decision boundaries at 0 and ± 2 on each axis.

QAM against PSK

At the same average energy, QAM spreads its points over the plane and PSK keeps them on a circle. Both distances come from the results above.

DISTANCE AT EQUAL ENERGY

$$d_{\mathrm{QAM}}^2 = \frac{6E_s}{M-1}, \qquad d_{\mathrm{PSK}}^2 = 4E_s \sin^2 \frac{\pi}{M}$$

Their ratio measures the advantage of QAM. The energy E_s cancels.

ADVANTAGE

$$R_M = \frac{d_{\text{QAM}}^2}{d_{\text{PSK}}^2} = \frac{6E_s/(M-1)}{4E_s \sin^2(\pi/M)} = \frac{3}{2(M-1)\sin^2(\pi/M)}$$

Above 1, QAM needs less energy for the same $d_{\min}$.

For $M = 16$, $\sin^2(\pi/16) = 0.0381$. So $R_{16} = 3/\big(30\sin^2(\pi/16)\big) = 2.63$, or $10\log_{10} 2.63 = 4.20$ dB.

The gap grows with M. QAM saves 1.65, 4.20, 7.02 and 9.95 dB over PSK at $M = 8$, 16, 32 and 64. PSK uses only a circle, and QAM uses the whole plane.

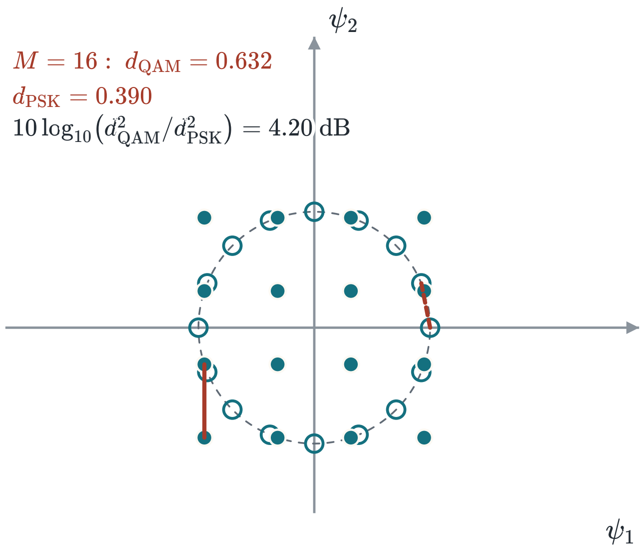


FIGURE 5.24 16-PSK (rings) and 16-QAM (dots) at the same $E_s = 1$. The dashed chord is the PSK $d_{\min}$ and the solid one the QAM $d_{\min}$.

QAM around us

- Wi-Fi 6 uses QAM with up to 1024 points, 10 bits a symbol. At $E_s = 1$ the points are $d_{\min} = \sqrt{6/1023} = 0.077$ apart.
- LTE and 5G NR use QAM with up to 256 points. Each axis then carries one of 16 levels, $I[n] \in \{\pm1, \pm3, \ldots, \pm15\}$.
- ADSL splits a phone line into tones 4.3125 kHz apart. Tone k carries b_k bits as a QAM set of 2^{b_k} points. Higher tones arrive weaker and carry fewer bits.
- Digital cable television uses 64-QAM and 256-QAM. Each $\mathbf{r} = \mathbf{s} + \mathbf{n}$ lands in a small cloud, here 64-QAM at $E_s/N_0 = 30$ dB.

- **Two PAM sets.** Square QAM puts one PAM set on each axis. The receiver decides each axis on its own.
- **Bits a symbol.** 256 points carry 8 bits a symbol and 1024 points carry 10.
- **Spacing.** At fixed energy $d_{\min}^2 = 6E_s/(M-1)$. Going from 256 to 1024 points costs 6.0 dB.

5.4 Frequency-shift keying and orthogonal signals

M-ary frequency-shift keying

M-ary FSK sends one of M tones, each orthogonal to the others. The spacing $1/(2T)$ keeps every pair orthogonal, as for BFSK.

M-FSK

$$s_m(t) = \sqrt{\frac{2E_s}{T}}\cos\big(2\pi(f_c + m\,\Delta f)\,t\big), \qquad \Delta f = \frac{1}{2T}$$

Each tone is one axis of an M-dimensional space, with its point at $\sqrt{E_s}$ on that axis.

Two points on different axes are at right angles. By Pythagoras their squared distance is $E_s + E_s$.

DISTANCES

$$\|\mathbf{s}_i - \mathbf{s}_j\|^2 = 2E_s, \qquad i \neq j$$

Every point is a neighbour of every other, so each has $M-1$ neighbours at one distance. 8-FSK has 7.

The union bound of Chapter 4 adds one Q term for each of the $M-1$ neighbours. Each term has $d^2/2N_0 = 2E_s/2N_0 = E_s/N_0$.

KEY RESULT · ORTHOGONAL SIGNALS

$$P_e \le (M-1)\,Q\left(\sqrt{\frac{E_s}{N_0}}\right), \qquad E_s = E_b \log_2 M$$

More tones add bits without bringing the points closer. The receiver picks the largest correlator output.

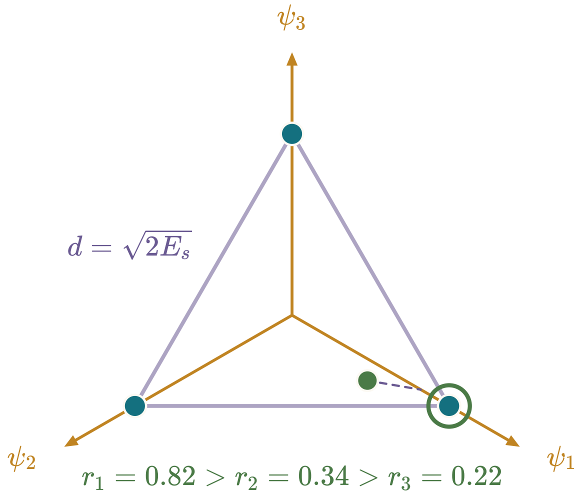


FIGURE 5.25 Three orthogonal signals, one on each axis at $\sqrt{E_s}$ from the origin. Every pair is $\sqrt{2E_s}$ apart. A received point with $r_1 > r_2 > r_3$ is decided as $\mathbf{s}_1$ (green ring).

More signals, less energy

A wrong symbol is any of the other $M-1$ with equal chance. Take one bit position of the k-bit label. It differs from the sent bit in 2^{k-1} of the 2^k labels, and the sent label is not one of them.

BITS FROM SYMBOLS

$$P_b = \frac{2^{k-1}}{2^k - 1} P_e$$

At $P_b = 10^{-5}$, $M = 2$ needs 12.6 dB and $M = 64$ about 6.1 dB. Each doubling of M lowers the E_b/N_0 needed.

⚠ A FLOOR

The gain has a limit. As M grows, $P_b \to 0$ only above $E_b/N_0 = \ln 2$, or -1.6 dB. More signals move the curves toward this limit, not past it. Chapter 6 shows why.

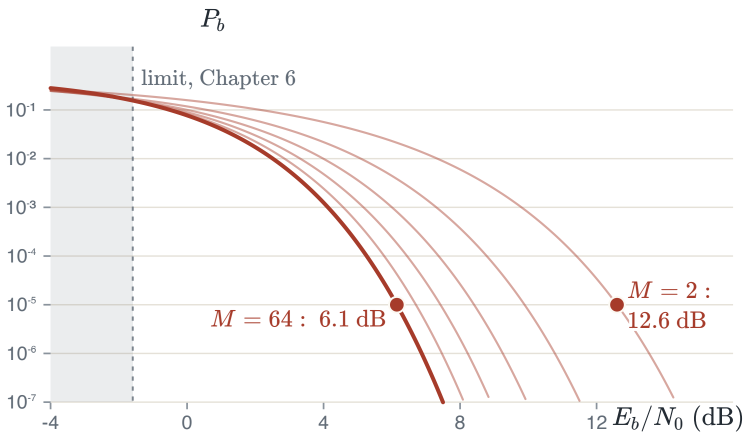


FIGURE 5.26 Exact bit error of M orthogonal signals for $M = 2$ to 64, with $M = 64$ drawn heavy. The curves move left as M grows. The shaded wall at -1.6 dB is the limit Chapter 6 derives.

Noncoherent FSK

A noncoherent receiver does not know the carrier phase. For each tone it uses a cosine and a sine correlator.

A tone of unit amplitude and unknown phase φ gives $y_c = \cos\varphi$ and $y_s = \sin\varphi$. The length of (y_c, y_s) is $\sqrt{\cos^2\varphi + \sin^2\varphi} = 1$, whatever φ is.

ENVELOPE DETECTOR

$$\ell_i = \sqrt{y_{c,i}^2 + y_{s,i}^2}$$

The envelope ℓ_i does not depend on φ. The receiver picks the larger ℓ_i.

Without the phase, the tones must stay orthogonal for every φ. Both the cosine and the sine correlations must then vanish.

$$\frac{1}{T}\int_0^T \cos(2\pi\Delta f\, t)\, dt = \frac{\sin(2\pi\Delta fT)}{2\pi\Delta fT}, \qquad \frac{1}{T}\int_0^T \sin(2\pi\Delta f\, t)\, dt = \frac{1-\cos(2\pi\Delta fT)}{2\pi\Delta fT}$$

Both are zero first at $2\pi\Delta fT = 2\pi$. This is twice the coherent spacing.

SPACING

$$\Delta f = \frac{1}{T}$$

KEY RESULT · NONCOHERENT BFSK

$$P_b = \frac{1}{2}e^{-E_b/2N_0}$$

At 10^{-5} it needs 13.4 dB against 12.6 dB coherent. The exponent holds $E_b/2N_0$ in place of E_b/N_0, so it needs twice the energy of binary DPSK.

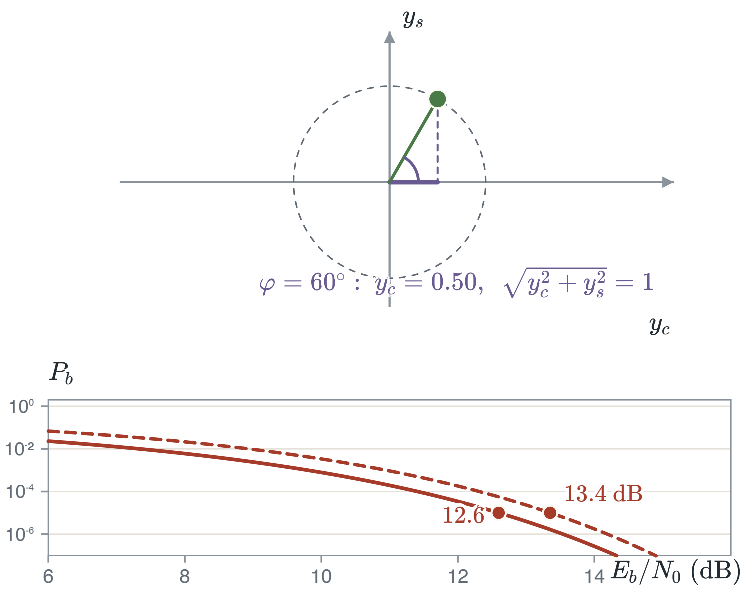


FIGURE 5.27 A tone with unknown phase $\varphi = 60^\circ$ gives $y_c = \cos\varphi$ and $y_s = \sin\varphi$, and its envelope stays 1 (top). Coherent (solid) and noncoherent (dashed) BFSK, 12.6 against 13.4 dB at 10^{-5} (bottom).

Frequency keying around us

- Caller ID and radio packet modems (Bell 202) send 1200 b/s as 1200 Hz for a one and 2200 Hz for a zero. The phase runs on across each switch.
- Bluetooth Low Energy sends 1 Mb/s with Gaussian FSK (GFSK). A Gaussian filter rounds each switch of frequency, so the spectrum falls off faster.
- GSM sends GMSK: MSK whose frequency pulse passes a Gaussian filter with $BT_b = 0.3$. The Gaussian pulse spreads over about three bits, where plain MSK holds one.
- Digital mobile radio (DMR) uses four-level FSK. Each symbol shifts the carrier by one of four offsets $\{\pm d, \pm 3d\}$ and carries 2 bits.
- **Frequency carries the bits.** FSK keeps a constant envelope. A cheap receiver can detect it without the carrier phase.

- **Smooth switching.** GFSK and GMSK smooth each change of frequency. The side lobes of the spectrum then fall off faster.
- **Band grows with M.** Orthogonal tones need a band that grows with M. Radio links therefore keep M at 2 or 4.

5.5 Bandwidth and the choice of scheme

Spectrum and bandwidth

Take PSK or QAM with a rectangular pulse of length T. The pulse has the Fourier transform $T\,\mathrm{sinc}(fT)$, so the spectrum of the signal is a sinc^2.

SPECTRUM

$$S(f) \propto T\,\mathrm{sinc}^2(fT), \qquad T = kT_b$$

Here $\mathrm{sinc}\,x = \sin(\pi x)/(\pi x)$. Longer symbols give a narrower spectrum, and the passband copy sits at $\pm f_c$.

The sinc^2 is first zero at $f = \pm 1/T$, so the main lobe is $2/T = 2R_b/k$ wide. At the same bit rate, QPSK symbols last $2T_b$. Its main lobe is R_b wide, half the $2R_b$ of BPSK.

A band W carries $2W$ dimensions a second. A scheme that uses N dimensions a symbol at $R_s = R_b/\log_2 M$ symbols a second needs $W = NR_s/2$. PAM on one sideband uses one dimension, PSK and QAM two, and orthogonal signals M.

BAND OF EACH FAMILY

$$\text{PAM (SSB)}: \frac{R_b}{2\log_2 M}, \qquad \text{PSK, QAM}: \frac{R_b}{\log_2 M}, \qquad \text{orthogonal}: \frac{MR_b}{2\log_2 M}$$

Orthogonal signals need M dimensions a symbol, so their band grows with M.

For example, 16-QAM at 10 Mb/s carries 4 bits a symbol, so $W = 10/4 = 2.5$ MHz.

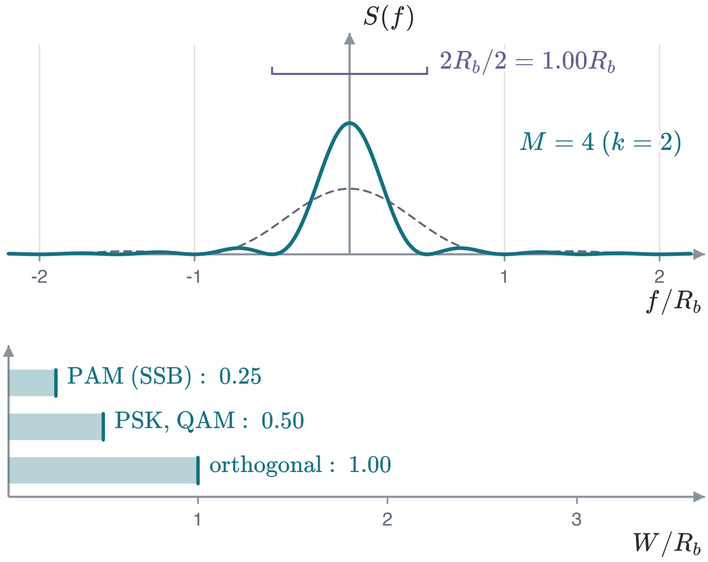


FIGURE 5.28 PSK or QAM with a rectangular pulse at a fixed bit rate, for $k = 2$ (solid) and $k = 1$ (dashed). The main lobe narrows to $2R_b/k$ (top). The band each family needs at $k = 2$ (bottom).

Offset QPSK and MSK

QPSK can swing through the origin when I and Q change sign at the same time. Its envelope then falls to zero.

> ⓘ OFFSET QPSK
>
> Delay Q by T_b. Only one axis changes at a time, so the path never crosses the origin and the envelope stays above $1/\sqrt{2}$.

Minimum-shift keying (MSK) goes further. Each bit shifts the frequency by $\pm\Delta f/2 = \pm 1/(4T_b)$, which turns the phase by $2\pi \cdot \frac{1}{4T_b} \cdot T_b = \pi/2$ over one bit.

> MSK
>
> $$\Delta\theta = \pm\frac{\pi}{2} \text{ a bit}, \qquad \Delta f = \frac{1}{2T_b}$$
>
> Each bit turns the phase by $\pm\pi/2$ at a steady rate. MSK is BFSK at the least orthogonal spacing, with a continuous phase.

Of the three, only MSK keeps a constant envelope. Its phase moves along the circle, so $|s(t)|$ never changes.

> ⚠ SPECTRUM
>
> The MSK main lobe reaches $0.75/T_b$ against $0.5/T_b$, so it is 50% wider. Its side lobes fall much faster.

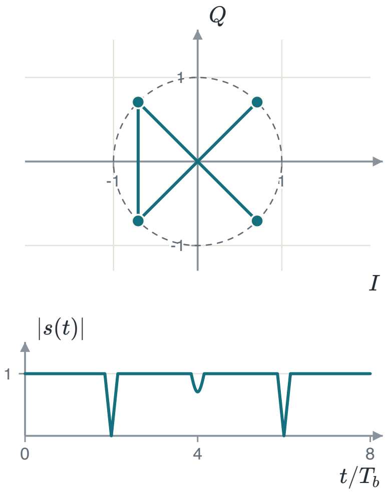


FIGURE 5.29 QPSK: the path of (I, Q) crosses the origin, and the envelope drops to zero.

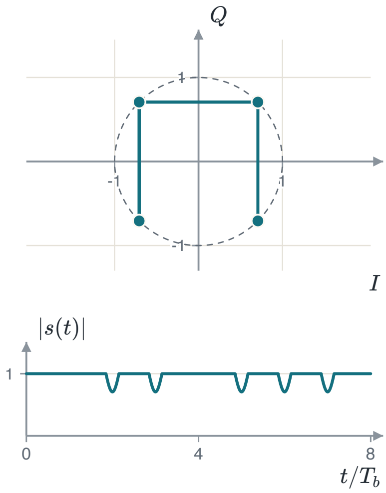


FIGURE 5.30 Offset QPSK: one axis moves at a time, and the envelope stays above $1/\sqrt{2}$.

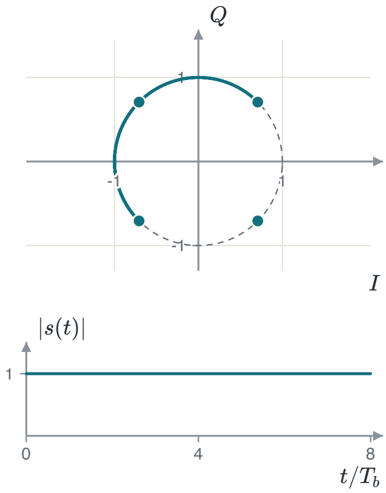


FIGURE 5.31 MSK: the path stays on the circle, and the envelope is constant.

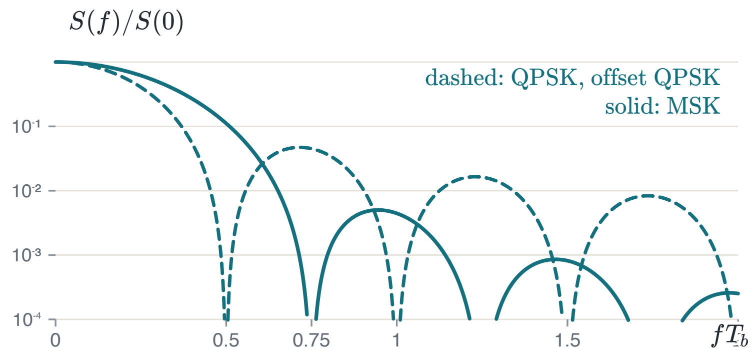


FIGURE 5.32 Spectra of QPSK and offset QPSK (dashed) and of MSK (solid). The MSK main lobe reaches $0.75/T_b$ against $0.5/T_b$, and its side lobes fall faster.

The bandwidth-efficiency plane

The bandwidth efficiency counts the bits a second that each hertz of band carries.

> BANDWIDTH EFFICIENCY
>
> $$r = \frac{R_b}{W}\ \mathrm{b/s/Hz}$$
>
> From the bands above, PSK and QAM have $r = \log_2 M$, PAM on one sideband $r = 2\log_2 M$, and orthogonal signals $r = 2\log_2 M/M$.

PAM, PSK and QAM raise r with M. Orthogonal signals lower it. Each family then sits on a plane of r against the E_b/N_0 it needs.

TABLE 5.1 The two regions of the bandwidth-efficiency plane.

REGION	FAMILIES	TRADE
$r > 1$: bandwidth-limited	PAM, PSK, QAM	More bits a symbol, more energy a bit.
$r < 1$: power-limited	Orthogonal signals	More band, less energy a bit.

⚠ A LIMIT

No scheme sits left of the curve $E_b/N_0 = (2^r - 1)/r$. Chapter 6 derives it.

For example, 64-QAM carries 6 bits a symbol, so $r = 6 > 1$. It is bandwidth-limited: it saves band and pays in energy.

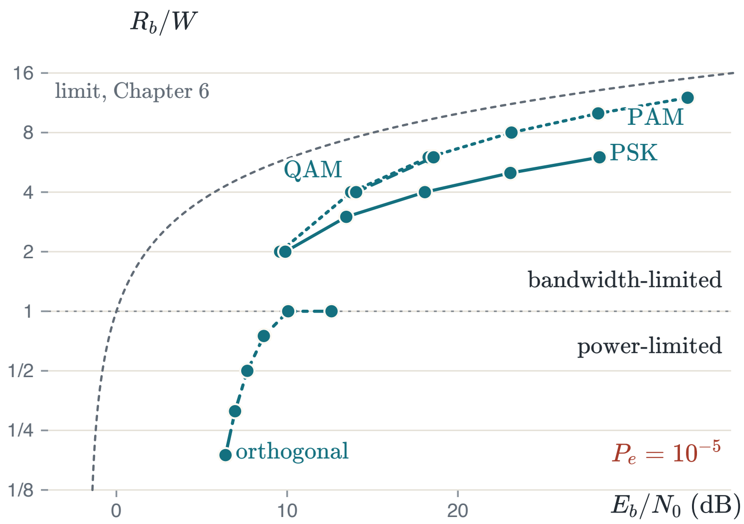


FIGURE 5.33 R_b/W against the E_b/N_0 each set needs for $P_e = 10^{-5}$, for M up to 64. The faint curve is the limit Chapter 6 derives.

Comparing the families

PAM, PSK and QAM are bandwidth-limited. They pay energy for each extra bit, and QAM pays the least of the three. Orthogonal signals are power-limited. They need less E_b/N_0 as M grows, at the price of band.

TABLE 5.2 Each family at $M = 16$ and $P_e = 10^{-5}$.

FAMILY	E_b/N_0	$r = R_b/W$	REGION
PAM (SSB)	23.1 dB	8	bandwidth-limited
PSK	18.1 dB	4	bandwidth-limited
QAM	14.0 dB	4	bandwidth-limited
Orthogonal (FSK)	7.7 dB	0.5	power-limited

The same 4 bits a symbol cost four very different amounts of energy. Sixteen orthogonal tones need about 7.7 dB, and 16-QAM about 14.0 dB.

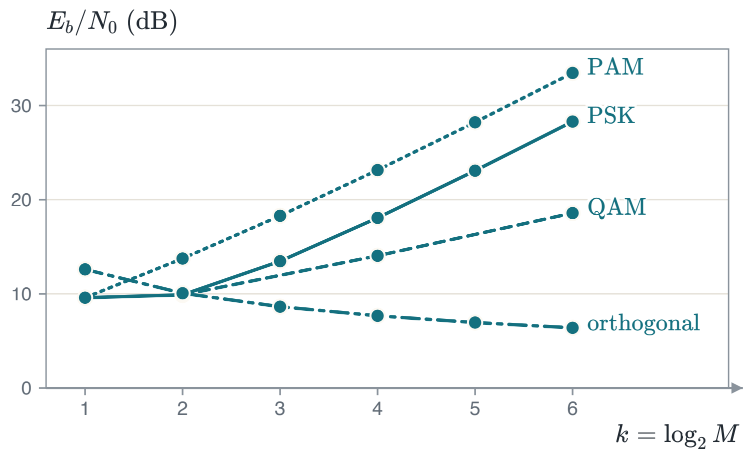


FIGURE 5.34 The E_b/N_0 each family needs for $P_e = 10^{-5}$, against the bits a symbol. PAM and PSK climb fastest, QAM more slowly, and orthogonal signals fall.

Adaptive modulation

Adaptive modulation uses this trade as the channel changes. The receiver measures E_s/N_0 and reports it. The transmitter picks the largest M that still meets the target bit error.

TABLE 5.3 The E_s/N_0 each set needs for $P_b = 10^{-5}$ with Gray labels.

SET	BITS A SYMBOL	E_s/N_0 FOR $P_b = 10^{-5}$
BPSK	1	9.6 dB
QPSK	2	12.6 dB
16-QAM	4	19.5 dB
64-QAM	6	25.6 dB
256-QAM	8	31.5 dB

Each step of two bits costs about 6 dB. The symbol rate stays fixed, so every step adds bits in the same band. A fading channel moves the link up and down the steps.

For example, a channel that gives $E_s/N_0 = 22$ dB clears the 16-QAM step at 19.5 dB but not 64-QAM at 25.6 dB. The link chooses 16-QAM.

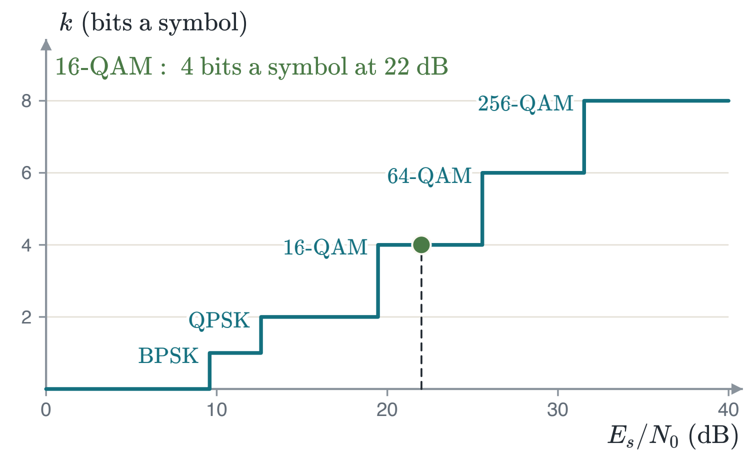


FIGURE 5.35 The largest set whose bit error stays at 10^{-5}, with Gray labels, against E_s/N_0. At 22 dB the link sends 16-QAM.

The receiver noise floor

A power in dBm is in decibels above 1 mW, $P_{\text{dBm}} = 10\log_{10}(P/1\ \text{mW})$. A ratio in dB adds to a power in dBm.

Every receiver hears thermal noise. At the reference temperature $T_0 = 290$ K its density is kT_0, with Boltzmann's constant $k = 1.38 \times 10^{-23}$ J/K. The receiver adds noise of its own, counted by the noise figure $F \geq 1$.

THERMAL NOISE

$$N_0 = kT_0F, \qquad kT_0 = (1.38 \times 10^{-23})(290) = 4.00 \times 10^{-21}\ \text{W/Hz}$$

$$10\log_{10}\frac{4.00 \times 10^{-21}\ \text{W/Hz}}{10^{-3}\ \text{W}} = -174\ \text{dBm/Hz}, \qquad \text{NF} = 10\log_{10} F$$

The noise figure in dB is NF. A receiver that adds no noise has $F = 1$, or $\text{NF} = 0$ dB.

⚠ ONE-SIDED DENSITY

kT_0F is N_0, not $N_0/2$. The two-sided density $N_0/2$ is 3 dB lower, -177 dBm/Hz with $F = 1$. It covers both signs of f, so both give $N = N_0B$.

A band of width B collects the noise power $N = N_0B$. In decibels the product becomes a sum.

NOISE POWER IN A BAND

$$N = N_0B, \qquad N_{\text{dBm}} = -174 + \text{NF} + 10\log_{10} B$$

For $B = 10$ MHz, $10\log_{10} 10^7 = 70$. With $\text{NF} = 5$ dB the noise floor is $N = -174 + 5 + 70 = -99$ dBm.

The noise floor grows as $10\log_{10} B$. A receiver that widens its band from 1 MHz to 10 MHz raises it by $10\log_{10} 10 = 10$ dB.

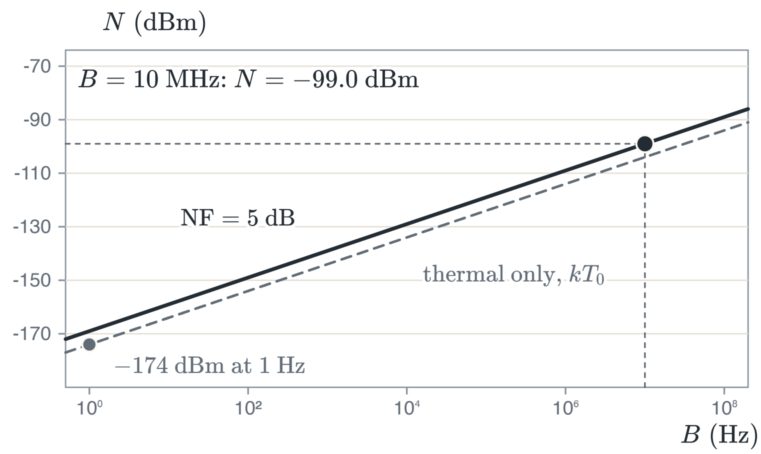


FIGURE 5.36 Noise power against the band. Dashed: thermal noise alone. Solid: a receiver with noise figure $\text{NF} = 5$ dB, which lies 5 dB higher. The guides read the floor at $B = 10$ MHz.

Sensitivity and the link budget

The sensitivity $P_{\min}$ is the least received power that meets the target error. The energy a bit is the received power times the bit time, $E_b = P_r/R_b$. So $E_b/N_0 = P_r/(R_b N_0)$. Solve for P_r at the required ratio.

$$P_{\min} = \left(\frac{E_b}{N_0}\right)_{\text{req}} R_b N_0 = \left(\frac{E_b}{N_0}\right)_{\text{req}} R_b\, kT_0 F$$

SENSITIVITY

$$P_{\min} = -174 + \mathrm{NF} + 10\log_{10} R_b + \left(\frac{E_b}{N_0}\right)_{\text{req,dB}} \text{dBm}$$

The band cancels. QPSK at 10 Mb/s needs $E_b/N_0 = 9.6$ dB for $P_b = 10^{-5}$. With $\mathrm{NF} = 5$ dB it needs $-174 + 5 + 70 + 9.6 = -89.4$ dBm.

The received power comes from the transmit power, the two antennas and the path. In free space the Friis formula gives it. The antenna gains G_t and G_r are measured against an antenna that radiates equally in every direction, and $\lambda = c/f_c$ is the wavelength.

$$P_r = P_t G_t G_r \left(\frac{\lambda}{4\pi d}\right)^2$$

Take $10\log_{10}$ of both sides. The products become sums, and the last factor becomes the path loss L_p.

FREE-SPACE BUDGET

$$P_r = P_t + G_t + G_r - L_p, \qquad L_p = 20\log_{10}\frac{4\pi d}{\lambda}$$

Powers are in dBm and antenna gains in dBi. At 2.4 GHz, $\lambda = 12.5$ cm and $L_p = 20\log_{10}(4\pi \cdot 1000/0.125) = 100.0$ dB at 1 km.

The loss L_p grows with d^2. Each doubling of the range raises it by $20\log_{10} 2 = 6$ dB, so P_r falls by 6 dB.

The margin $P_r - P_{\min}$ covers fading and losses the free-space model leaves out. The range of a link is the largest d at which the margin is still met.

Take $P_t = 20$ dBm and two 10 dBi antennas at 2.4 GHz. At 1 km, $P_r = 20 + 10 + 10 - 100.0 = -60.0$ dBm, which is 29.4 dB above the QPSK sensitivity of -89.4 dBm. Keeping a 10 dB margin allows $L_p = 119.4$ dB, a range of 9.3 km.

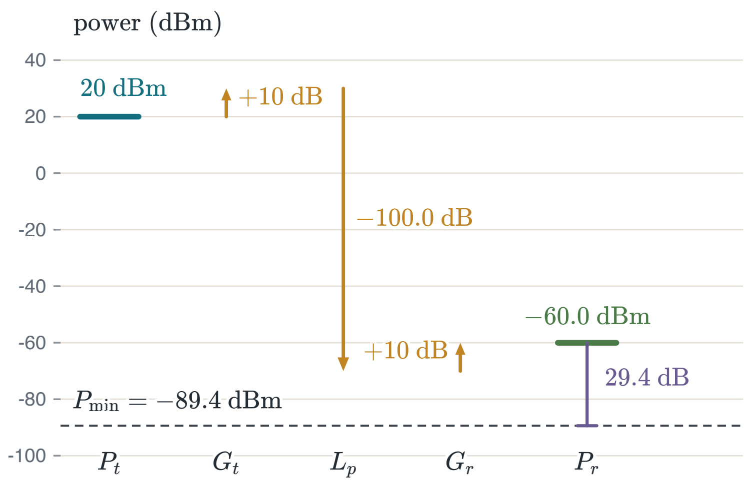


FIGURE 5.37 The budget at $d = 1$ km: $P_t = 20$ dBm, two 10 dBi antennas and the path loss at 2.4 GHz, against the sensitivity of QPSK at 10 Mb/s. The violet bracket is the margin.

EXAMPLE 5.5 — A LINK BUDGET

GIVEN A 5.8 GHz link has a band $B = 25$ MHz with roll-off $\alpha = 0.25$. It has $P_t = 20$ dBm, $G_t = G_r = 12$ dBi and $\mathrm{NF} = 7$ dB, and it keeps a 15 dB margin. For $P_b = 10^{-5}$, QPSK needs $E_b/N_0 = 9.6$ dB and 16-QAM 13.4 dB.

FIND The bit rate, the sensitivity and the range of each scheme. Which reaches farther?

METHOD The band fixes the symbol rate $R_s = B/(1+\alpha)$. The sensitivity follows from R_b and NF. The allowed loss is $L_p = P_t + G_t + G_r - P_{\min} - \text{margin}$, and inverting $L_p = 20\log_{10}(4\pi d/\lambda)$ gives $d = (\lambda/4\pi)\,10^{L_p/20}$.

SOLUTION $R_s = 25/1.25 = 20$ Msym/s, so QPSK carries 40 Mb/s and 16-QAM 80 Mb/s. Then $10\log_{10}(40\times10^6) = 76.0$ and $10\log_{10}(80\times10^6) = 79.0$, and $-174 + \mathrm{NF} = -167$ dBm/Hz. QPSK: $P_{\min} = -167 + 76.0 + 9.6 = -81.4$ dBm and $L_p = 20 + 24 + 81.4 - 15 = 110.4$ dB. 16-QAM: $P_{\min} = -167 + 79.0 + 13.4 = -74.6$ dBm and $L_p = 20 + 24 + 74.6 - 15 = 103.6$ dB. With $\lambda = 3\times10^8/5.8\times10^9 = 5.17$ cm, $\lambda/4\pi = 4.12\times10^{-3}$ m. So QPSK reaches $d = 4.12\times10^{-3}\cdot10^{110.4/20} = 1.36$ km and 16-QAM $d = 4.12\times10^{-3}\cdot10^{103.6/20} = 0.62$ km. QPSK reaches farther.

CHECK The two sensitivities differ by $-74.6 - (-81.4) = 6.8$ dB: 3.8 dB of E_b/N_0 and 3.0 dB of bit rate. A loss 6.8 dB smaller is a range $10^{6.8/20} = 2.19$ times shorter, and $1.36/0.62 = 2.19$.

⊗ COMMON ERROR

Use $20\log_{10}$, not $10\log_{10}$, to turn a loss into range. A loss of 6.8 dB is a factor 2.2 in range, not 4.8. The loss L_p grows with d^2.

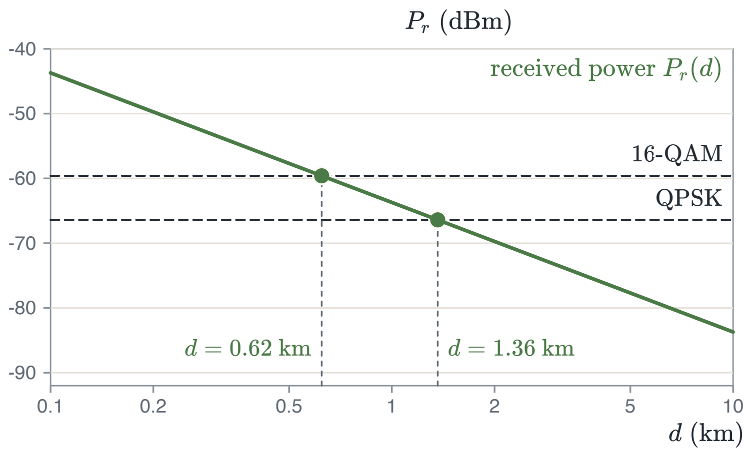


FIGURE 5.38 Example 5.5: received power against distance, with each sensitivity plus the 15 dB margin. Where they cross is the range.

Choosing a scheme around us

- Satellite DVB-S2 also sends 16APSK: 4 points on an inner ring and 12 on an outer ring. Two amplitudes suit a satellite amplifier better than the three of 16-QAM.
- Bluetooth Classic sends 1 million symbols a second. GFSK carries 1 bit a symbol, $\pi/4$-DQPSK carries 2 and 8DPSK 3, for 1, 2 and 3 Mb/s.
- GSM sends GMSK, whose envelope stays constant. A 16-QAM envelope jumps between three levels, so its amplifier must stay linear.
- LTE and 5G NR change the modulation as the channel changes, from QPSK with $k = 2$ up to 256-QAM with $k = 8$ bits a symbol.

- **Energy or band.** A link short of band packs more bits into each symbol. A link short of energy spreads them over orthogonal signals.
- **Amplifier cost.** A constant envelope lets the amplifier run near saturation. That matters in a handset and on a satellite.
- **Adapt.** Most modern links measure the channel and change M as it changes.

5.6 Summary

From bits to decisions

A link carries bits to decisions in four moves.

1. Map k bits to a point of the set.
2. The IQ modulator puts the point on the carrier.
3. Two correlators turn $r(t)$ into $\mathbf{r}$.
4. The receiver picks the nearest point.

Here the error rate follows from E_b/N_0 and the set. Chapter 6 asks how many bits a second a band and a power can carry at all.

The number of bits a second is the symbol rate times $k = \log_2 M$. For example, 16-QAM at 10^6 symbols a second carries $k = \log_2 16 = 4$ bits a symbol, so 4×10^6 b/s.

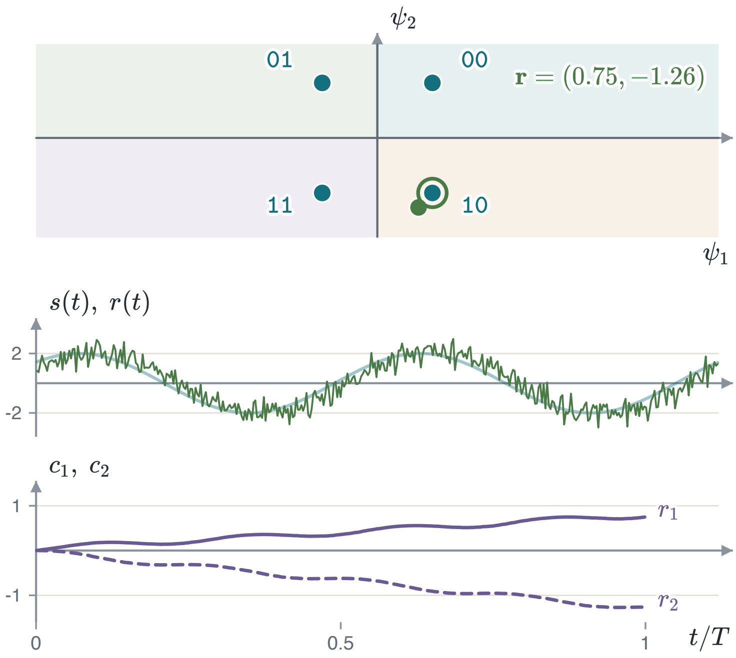


FIGURE 5.39 The bits 10 through a QPSK link. Top: the point $(1, -1)$ and the received point. Middle: the carrier $s(t)$ (faint) and the noisy $r(t)$. Bottom: the two correlator outputs, which end at r_1 and r_2.

Quick check

1 BPSK reaches $P_b = 10^{-5}$ at 9.6 dB. Where does BFSK reach it?

Answer: 12.6 dB. Half the squared distance costs 3 dB.

2 How many bits does each 8-PSK symbol carry?

Answer: 3 bits, since $\log_2 8 = 3$.

3 4-ASK moves to 8-ASK at the same $d_{\min}$. By about how much does E_s grow?

Answer: About 6 dB. $E_s \propto M^2 - 1$, so $10\log_{10}(63/15) = 6.2$ dB.

4 Against what does a DPSK receiver decide each symbol?

Answer: The previous symbol. The bit is in the change of phase from the symbol before.

5 Orthogonal signals at a fixed bit rate go from $M = 8$ to $M = 16$. What does the band do?

Answer: It widens. $W = MR_b/(2\log_2 M)$ goes from $1.33R_b$ to $2R_b$.

6 At a fixed bit rate, how wide is the QPSK main lobe next to that of BPSK?

Answer: Half as wide. Its symbols last twice as long.

Results of the chapter

TABLE 5.4 Summary of Chapter 5: the results the chapter carries forward.

QUESTION	RESULT	ANCHOR
What does a carrier do to the spectrum?	$U(f) = \frac{1}{2}S(f - f_c) + \frac{1}{2}S(f + f_c)$. The band doubles to $2W$ and the energy halves.	PS CH8.5.1
What does the IQ modulator send?	$s(t) = I\cos(2\pi f_c t) - Q\sin(2\pi f_c t)$: one point of the plane a symbol.	PS CH8.6, 8.7
How do the binary schemes compare?	BPSK: $Q\left(\sqrt{2E_b/N_0}\right)$. BFSK and BASK: $Q\left(\sqrt{E_b/N_0}\right)$, 3 dB worse.	PS CH8.3.3, 8.6.1, 9.5
What is the M-PSK neighbour distance?	$d_{\min} = 2\sqrt{E_s}\sin(\pi/M)$, and $P_e \approx 2Q\left(\sqrt{2E_s/N_0}\sin(\pi/M)\right)$.	PS CH8.6.1, 8.6.3
What do Gray labels buy?	Neighbours differ in one bit, so $P_b \approx P_e/\log_2 M$.	PS CH8.6.1, 8.6.3
What does DPSK save, and what does it cost?	No carrier phase. $P_b = \frac{1}{2}e^{-E_b/N_0}$, under 1 dB behind BPSK.	PS CH8.6.4, 8.6.5
What is the M-ASK error?	$\frac{2(M-1)}{M}Q\left(\sqrt{6E_s/((M^2-1)N_0)}\right)$, about 6 dB a doubling.	PS CH8.5.3
What is the square-QAM error?	$4(1 - 1/\sqrt{M})\,Q\left(\sqrt{3E_s/((M-1)N_0)}\right)$, with $E_s = (M-1)d^2/6$.	PS CH8.7.3
How much does QAM save over PSK?	4.20 dB at $M = 16$ and 9.95 dB at $M = 64$.	PS CH8.7.3
What do M orthogonal signals give?	$M - 1$ neighbours at $\sqrt{2E_s}$. The E_b/N_0 needed falls as M grows, toward -1.6 dB.	PS CH9.1.1, 9.1.2
What does noncoherent BFSK need?	Tones $1/T$ apart and an envelope detector. $P_b = \frac{1}{2}e^{-E_b/2N_0}$.	PS CH9.5.2, 9.5.3
What is MSK?	BFSK at $\Delta f = 1/(2T_b)$ with a continuous phase. Its envelope is constant.	PS CH9.6.1, 9.6.2
How wide is each family?	PSK and QAM: $R_b/\log_2 M$. Orthogonal: $MR_b/(2\log_2 M)$.	PS CH10.2, 9.7
Where does each family sit on the plane?	PAM, PSK and QAM: bandwidth-limited, $r > 1$. Orthogonal: power-limited, $r < 1$.	PS CH9.7

Pick the point set, read $d_{\min}$ and $\bar{N}_{\min}$, and apply the receiver of Chapter 4. Chapter 6 asks how far any scheme can go.

Projects to try

Four optional projects use the chapter on simulated signals. Each gives an aim, what it practises, a few steps and what to look for.

AN IQ MODEM FROM SAMPLES

Build a QPSK and 16-QAM modem from sampled carriers and measure how often it is wrong.

Practises: The IQ modulator as two products and a sum. Two correlators as sums over samples. Counting errors to estimate P_e.

1. Build a cosine and a sine carrier over one symbol at 64 samples.
2. Map random bits to QPSK points and form $s(t) = I\cos - Q\sin$.
3. Add Gaussian noise, correlate, and pick the nearest point.
4. Repeat for 16-QAM and plot both error rates against E_b/N_0.

Look for: the measured points follow the formulas of this chapter once enough errors are counted.

GRAY AGAINST NATURAL LABELS

Measure how much Gray labels lower the bit error of 8-PSK.

Practises: Gray code as $i \oplus \lfloor i/2 \rfloor$. Why a symbol error usually lands on a neighbour. Bit error against symbol error.

1. Send random 8-PSK symbols over noise.
2. Label the points once with natural binary and once with Gray code.
3. Count the bit errors of each on the same noisy symbols.
4. Plot both bit errors and $P_e/3$ against E_b/N_0.

Look for: the Gray curve sits on $P_e/3$. The natural curve sits above it.

DPSK WITH A DRIFTING PHASE

See why differential detection survives a slowly drifting carrier phase.

Practises: Differential encoding of bits into phase changes. Coherent against differential detection. What a phase drift does to each.

1. Encode random bits as binary DPSK.
2. Turn each received symbol by a phase that grows slowly with time.
3. Detect coherently with a fixed reference and differentially.
4. Count the errors of each as the drift rate grows.

Look for: the coherent receiver fails once the drift passes $90°$. The differential one barely notices.

AN ADAPTIVE LINK

Build a link that changes its constellation as the SNR changes.

Practises: Thresholds for a target bit error. Throughput against a fixed scheme. Why a margin above each threshold helps.

1. Draw an SNR that wanders slowly between 0 and 35 dB.
2. At each block pick the largest set that meets the target.
3. Simulate the block and count its errors.
4. Compare the bits delivered with a fixed QPSK link.

Look for: the adaptive link carries several times the bits at about the same error rate.

CHAPTER

6

An introduction to information theory

Two numbers set what any code can do. A source has an entropy: the fewest bits a symbol that any lossless code can reach. A channel has a capacity: the most bits a use that any code can carry reliably.

6.1 Information and entropy

A source emits one symbol at a time from an alphabet $s_1, \ldots, s_K$. Symbol s_k appears with probability p_k.

Self-information

The **self-information** of a symbol measures the information it carries as minus the logarithm of its probability.

SELF-INFORMATION

$$I(s_k) = \log_2 \frac{1}{p_k} = -\log_2 p_k \quad \text{bits}$$

A symbol of probability p_k carries $I(s_k)$ bits when it arrives. Base 2 gives bits, and base e gives nats.

A symbol of probability $1/8$ carries $-\log_2 \frac{1}{8} = \log_2 8 = 3$ bits: three halvings from certainty. The figure plots $I(p)$ against p.

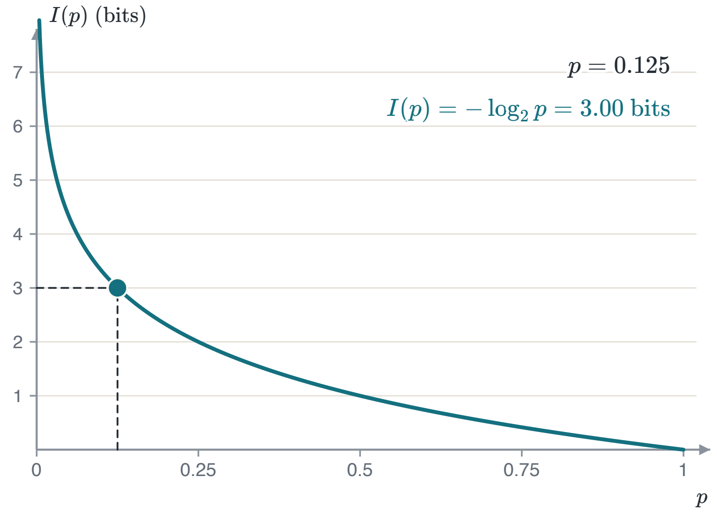


FIGURE 6.1 Self-information against probability, shown at $p = 1/8$, where $I = 3$ bits. A rare symbol carries many bits, and a certain one carries none.

THREE PROPERTIES

$I(s_k) \geq 0$, and $I(s_k) = 0$ when $p_k = 1$. A certain symbol tells nothing.
Rarer symbols carry more: $I(s_k) > I(s_j)$ when $p_k < p_j$.
Independent symbols add. Their probabilities multiply, and the logarithm turns the product into a sum: $I(s_j s_k) = I(s_j) + I(s_k)$.

COMMON ERROR

Write $-\log_2 p_k$, not $\log_2 p_k$. The logarithm of a probability is negative, and information is not.

Information as yes/no questions

A bit has a plain reading as one yes/no question. A question whose two answers are equally likely halves the candidates. One such answer gives one bit.

EQUALLY LIKELY CASES

$$K = 2^m \text{ cases} \implies m = \log_2 K \text{ questions}$$

Eight equally likely cards take $\log_2 8 = 3$ questions: the answers cut 8 to 4, 4 to 2 and 2 to 1. Sixteen outcomes take $\log_2 16 = 4$ questions, which halve 16 to 8, 4, 2 and 1.

A skewed source is searched faster on average by asking about the likely symbol first. Take four symbols with probabilities $\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{8}$. The first question asks for s_1. The second asks for s_2, and the third separates s_3 from s_4.

AVERAGE NUMBER OF QUESTIONS

$$\begin{aligned} \bar{L} &= \tfrac{1}{2}(1) + \tfrac{1}{4}(2) + \tfrac{1}{8}(3) + \tfrac{1}{8}(3) \\ &= 0.5 + 0.5 + 0.375 + 0.375 \\ &= 1.75 = H \end{aligned}$$

Symbol s_k is found after $-\log_2 p_k$ questions: 1, 2, 3 and 3 here. So the average number of questions is the average self-information of the source. That average is the entropy H, defined next.

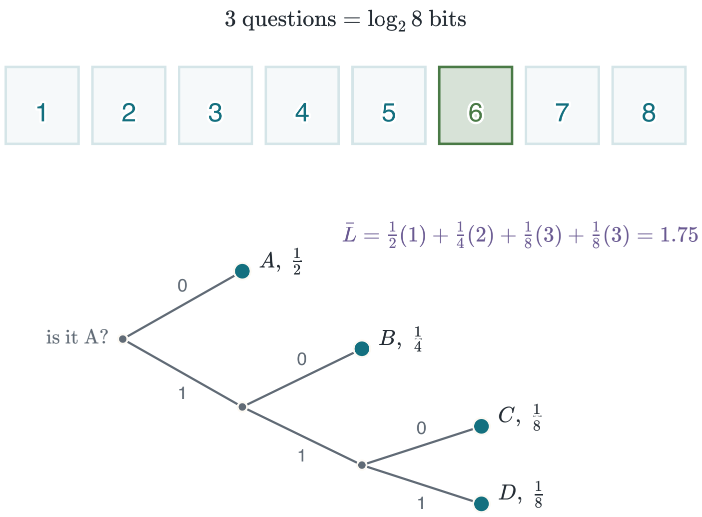


FIGURE 6.2 Top: one of eight equally likely cards found with three yes/no questions. Bottom: the skewed source $\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{8}$ asked with a tree, likely symbol first. Shown at the last step, where the average is $\bar{L} = 1.75$.

Entropy

The **entropy** of a source is the average self-information of its symbols. Each symbol's information is weighted by how often the symbol occurs.

ENTROPY

$$H(S) = \sum_{k=1}^{K} p_k I(s_k) = -\sum_{k=1}^{K} p_k \log_2 p_k$$

The average information of a symbol, in bits a symbol.

For the source $0.7, 0.2, 0.1$, write out the three terms and add them.

ENTROPY OF A THREE-SYMBOL SOURCE

$$\begin{aligned} H(S) &= -0.7\log_2 0.7 - 0.2\log_2 0.2 - 0.1\log_2 0.1 \\ &= 0.360 + 0.464 + 0.332 \\ &= 1.157 \text{ bits} \end{aligned}$$

The figure draws both quantities for each symbol: the bits it carries, and that number weighted by its probability.

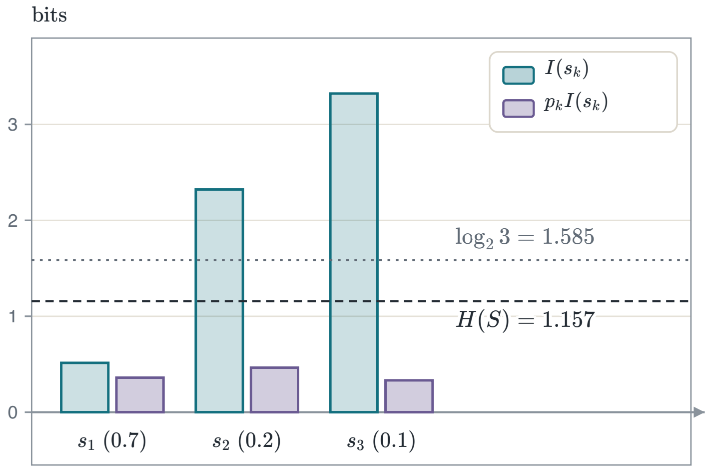


FIGURE 6.3 For each symbol of $0.7, 0.2, 0.1$: the bits it carries (left bar) and its weighted share (right bar). The weighted bars add to $H(S) = 1.157$, below $\log_2 3 = 1.585$.

The entropy lies between two bounds.

ENTROPY BOUNDS

$$0 \leq H(S) \leq \log_2 K$$

The lower bound holds when one symbol has probability 1 and nothing is in doubt. The upper bound holds when all K symbols are equally likely. With $p_k = 1/K$ the sum is $\sum_k \frac{1}{K} \log_2 K = \log_2 K$. Three equal symbols give $\log_2 3 = 1.585$ bits, more than the 1.157 of the skewed source.

The binary entropy function

A source of two symbols with probabilities p and $1 - p$ has the **binary entropy function**.

BINARY ENTROPY

$$H_b(p) = -p \log_2 p - (1 - p) \log_2(1 - p)$$

It is symmetric about $p = \frac{1}{2}$, where $H_b = 1$ bit. It is 0 only at $p = 0$ and $p = 1$.

The top of the curve is flat: at $p = \frac{1}{2}$, $H_b = 1$ bit, and it falls slowly near the top. At $p = 0.11$ the two terms are $-0.11 \log_2 0.11 = 0.350$ and $-0.89 \log_2 0.89 = 0.150$. So $H_b(0.11) = 0.500$ bit.

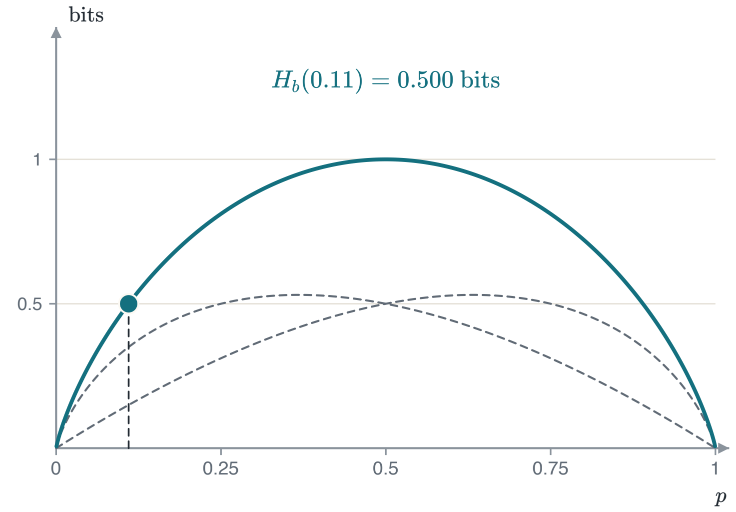


FIGURE 6.4 The binary entropy function (solid) and its two terms (dashed), shown at $p = 0.11$. The top is flat near $p = \frac{1}{2}$.

A **binary symmetric source** emits equally likely, independent bits. Each bit carries one bit of information, and no lossless code can shorten the stream.

A sampled source produces symbols at a fixed rate. Its **information rate** is the entropy of a sample times the number of samples a second.

EXAMPLE 6.1 — THE INFORMATION RATE OF A SOURCE

GIVEN: A source band-limited to $W = 3$ kHz is sampled at the Nyquist rate. Each sample takes one of four levels with probabilities $0.4, 0.3, 0.2, 0.1$.

FIND: The information rate R in bits a second.

METHOD: The rate is bits a sample times samples a second, $R = H(S)\, f_s$. The Nyquist rate of Chapter 1 is $f_s = 2W = 6000$ samples a second.

SOLUTION: $H(S) = -\sum_k p_k \log_2 p_k = 0.529 + 0.521 + 0.464 + 0.332 = 1.846$ bits a sample. Then $R = 1.846 \times 6000 = 11\,079$ b/s.

CHECK: Four levels carry at most $\log_2 4 = 2$ bits a sample, or $12\,000$ b/s. The skewed levels carry less, as the entropy bound requires.

COMMON ERROR

Multiply by the sample rate $2W$, not by the bandwidth W. Using 3000 samples a second gives half the rate, 5539 b/s.

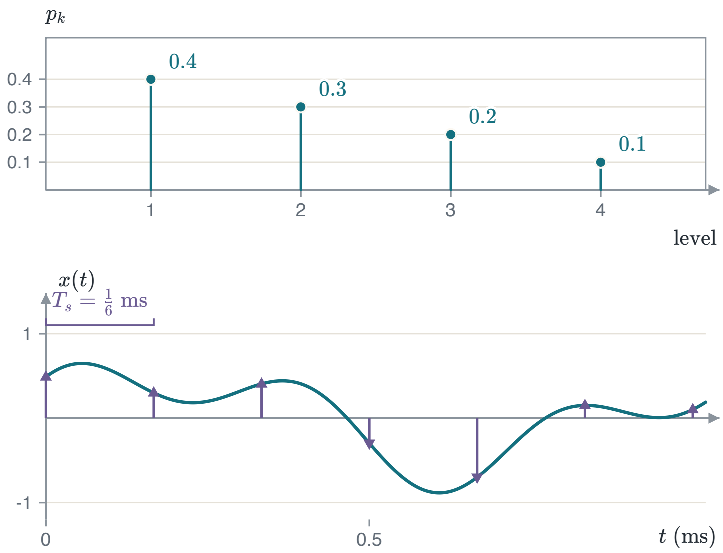


FIGURE 6.5 Top: the four levels of Example 6.1 and their probabilities. Bottom: a source band-limited to 3 kHz, sampled every $T_s = \frac{1}{6}$ ms, which is 6000 samples a second.

Extended sources

A coder can also take the symbols in blocks of n. Each block is one symbol of a new source, the **n-th extension** S^n, whose alphabet has K^n symbols.

The entropy of a block follows from the additive property. Write the entropy of a pair, split the logarithm of the product, and sum out the other symbol.

ENTROPY OF A PAIR

$$\begin{aligned} H(S^2) &= -\sum_{i,j} p_i p_j \log_2(p_i p_j) \\ &= -\sum_{i,j} p_i p_j \left(\log_2 p_i + \log_2 p_j\right) \\ &= -\sum_i p_i \log_2 p_i \sum_j p_j - \sum_j p_j \log_2 p_j \sum_i p_i \\ &= H(S) + H(S) \end{aligned}$$

Each inner sum of probabilities is 1. Repeating the step for n symbols gives the general rule.

EXTENSION

$$H(S^n) = n\,H(S)$$

The symbols of a memoryless source are independent, so their information adds. For $0.7, 0.2, 0.1$ the nine pair probabilities are $0.49, 0.14, 0.07, 0.14, 0.04, 0.02, 0.07, 0.02, 0.01$. Summed the long way they give $H(S^2) = 2.3136 = 2 \times 1.1568$ bits. Triples give $H(S^3) = 3 \times 1.1568 = 3.470$ bits.

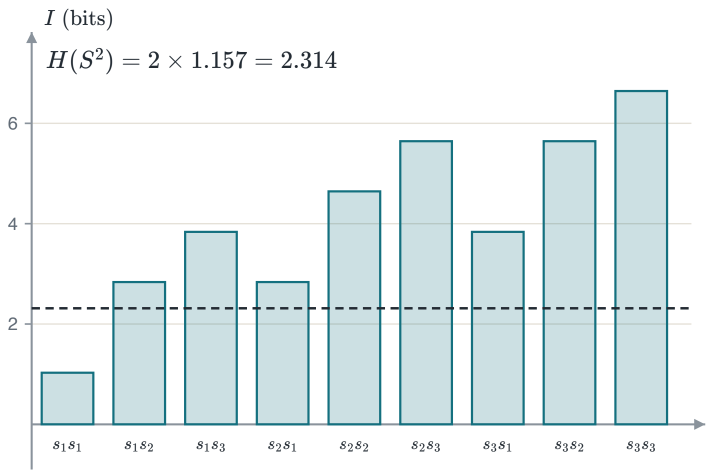


FIGURE 6.6 The bits each pair of S^2 carries, for the source $0.7, 0.2, 0.1$. A pair carries the sum of its two symbols' bits, so the average (dashed) doubles to 2.314.

> ⚠ MEMORY
>
> The rule needs a memoryless source. With memory, as in English where q is followed by u, a block carries less than $nH(S)$. Compressors live on that surplus.

Entropy around us

To find the entropy of real data, count how often each symbol occurs, then apply $H = -\sum_k p_k \log_2 p_k$. Four everyday sources:

- The 24 symbols of a short speech, letters and space, by frequency. $H = -\sum_k p_k \log_2 p_k = 3.97$ bits a symbol, against $\log_2 24 = 4.58$.
- A fair die carries $\log_2 6 = 2.585$ bits a throw. A die loaded to show six half the time carries $H = 2.161$ bits.
- A scanned text line, 8 pixels a millimetre, is 90% white. As independent pixels it carries $H_b(0.1) = 0.469$ bit a pixel, not 1.
- A drawn 64×64 grey image, 8 levels. Its histogram gives $H = 2.42$ bits a pixel, below $\log_2 8 = 3$.

> ✓ UNEVEN IS CHEAPER
>
> Each source here sits below $\log_2 K$. The gap is what a variable-length code can save.

> ⚠ MEMORY
>
> Letters and pixels depend on their neighbours. The entropy with memory is lower still, about 1.3 bits a letter for English.

6.2 The limits of compression

Code length

A **source encoder** maps each symbol s_k to a string of bits, its **codeword**, of length l_k. Common symbols should get short codewords and rare symbols long ones.

AVERAGE LENGTH AND EFFICIENCY

$$\bar{L} = \sum_{k=1}^{K} p_k l_k, \qquad \eta = \frac{H(S)}{\bar{L}} \leq 1$$

The code for $\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{8}$ with lengths $1, 2, 3, 3$ has $\bar{L} = \frac{1}{2}(1) + \frac{1}{4}(2) + \frac{1}{8}(3) + \frac{1}{8}(3) = 1.75$. That equals H, so $\eta = 1$.

In this code each length equals the information of its symbol. The figure sets the two side by side.

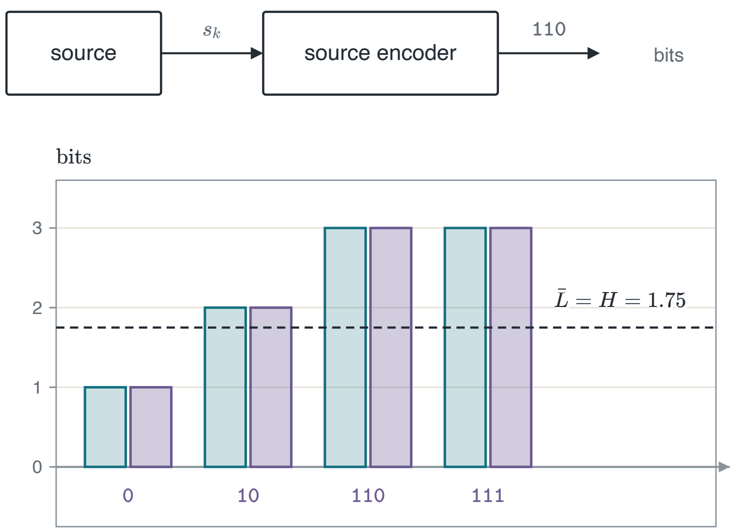


FIGURE 6.7 The source encoder maps each symbol to a codeword. For $\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{8}$ each code length (right bar) equals the information of its symbol (left bar).

SOURCE CODING THEOREM

Every uniquely decodable code has $\bar{L} \geq H(S)$. The entropy is the fewest bits a symbol that a lossless code can reach.

English shows how far a simple code can sit from this limit. With its memory, English carries about 1.3 bits a letter. A letter-by-letter code needs 4.22 bits, so $\eta = 1.3/4.22 = 0.31$.

Typical sequences

Long sequences show where the limit comes from. Take n bits from a binary source with $P(1) = p$. A long sequence has about np ones and $n(1-p)$ zeros.

The probability of such a sequence follows from these counts. Take its logarithm and divide by $-n$.

PROBABILITY OF A TYPICAL SEQUENCE

$$P(\mathbf{x}) = p^{np}(1-p)^{n(1-p)}$$
$$-\tfrac{1}{n}\log_2 P(\mathbf{x}) = -p\log_2 p - (1-p)\log_2(1-p) = H$$
$$P(\mathbf{x}) = 2^{-nH}$$

All such sequences have nearly the same probability, 2^{-nH}. They are the **typical sequences**.

A sequence counts as typical when its value of $-\frac{1}{n}\log_2 P$ lies within ϵ of H. For large n the typical sequences hold almost all the probability. Their probabilities then add to about one, so there are about 2^{nH} of them.

THE TYPICAL SET

$$P(\mathbf{x}) \approx 2^{-nH}, \qquad |A| \approx 2^{nH} \text{ of } 2^n \text{ sequences}$$

The rest are rare. With $p = 0.2$, $H = 0.722$. At $n = 100$ there are about $2^{100\times 0.722} = 2^{72.2}$ typical sequences, a tiny fraction of the 2^{100}.

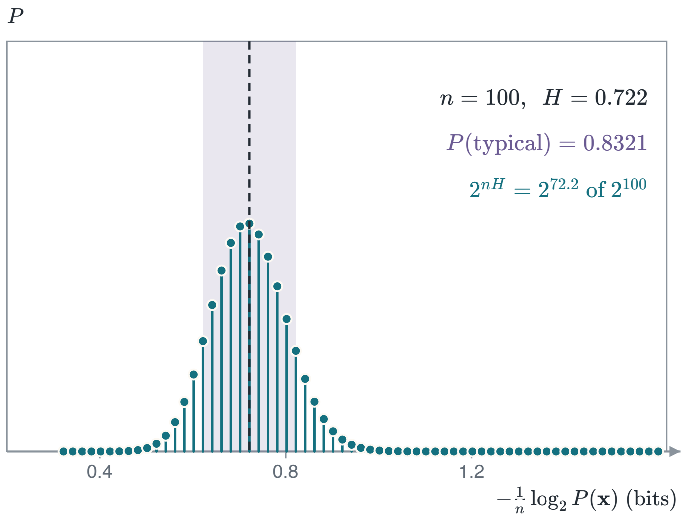


FIGURE 6.8 Every sequence of n bits with $P(1) = 0.2$, placed at its value of $-\frac{1}{n}\log_2 P$ with its probability, shown at $n = 100$. The probability gathers in the band $H \pm 0.1$ (shaded).

⚠ UNIFORM SOURCE

At $p = \frac{1}{2}$, $H = 1$ and every one of the 2^n sequences is typical. There is nothing to compress.

The source coding theorem

A code can give each typical sequence a number, its index, and give up on the rare rest. About nH bits name a typical sequence, which is H bits a symbol. The chance of meeting a rare sequence falls to zero as n grows.

SOURCE CODING THEOREM

$$R > H: \; P_{\text{error}} \to 0, \qquad R < H: \; P_{\text{error}} \not\to 0$$

A code of R bits a symbol can be made reliable when R exceeds H, and never when R falls below it. A source with $H = 0.5$ bit a symbol, in blocks of $n = 1000$, needs about $nH = 500$ bits a block to name its 2^{500} typical blocks.

Ten bits show the count in full. Take $p = 0.2$ and $\epsilon = 0.1$, so the band runs from 0.622 to 0.822. A sequence on the edge of the band counts as typical.

THE TYPICAL SEQUENCES OF TEN BITS

$$k = 2: \quad -\tfrac{1}{10}\log_2\left(0.2^2\,0.8^8\right) = 0.722$$

$$k = 1: \; 0.522, \qquad k = 3: \; 0.922$$

Here k is the number of ones. Only the sequences with two ones fall inside the band. There are $\binom{10}{2} = 45$ of them, each of probability $0.2^2\,0.8^8 = 0.00671$. Together they hold $45 \times 0.00671 = 0.3020$ of the probability. An index of 6 bits names them, since $2^6 = 64 \geq 45$.

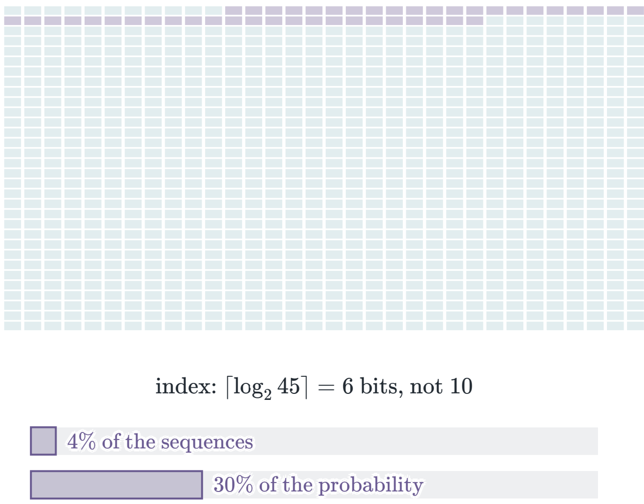


FIGURE 6.9 All 1024 sequences of ten bits with $P(1) = 0.2$, in order of their number of ones, shown at the last step. The 45 typical ones are 4% of the sequences, hold 30% of the probability and take a 6-bit index.

The share held by the typical set grows with n.

TABLE 6.1 Probability held by the typical set, for $p = 0.2$ and $\epsilon = 0.1$.

n	$P(\text{typical})$
10	30%
100	83%
1000	99.99%

Prefix codes

Short codewords help only if the receiver can split the bit stream back into codewords. A code is **uniquely decodable** when every bit string of the code comes from one symbol string only.

A **prefix code** is one in which no codeword starts another. The decoder names a symbol as soon as its last bit arrives, so a prefix code is also called **instantaneous**.

TABLE 6.2 Three codes for the same four symbols.

SYMBOL	CODE I	CODE II	CODE III
s_1	0	0	0
s_2	1	10	01
s_3	00	110	011
s_4	11	111	0111

Code I is not uniquely decodable: 00 is s_3 or $s_1 s_1$. Code II is a prefix code. It splits the bits 0101100111 as 0 10 110 0 111, which is $s_1 s_2 s_3 s_1 s_4$.

Code III is uniquely decodable, because every codeword starts with 0. It is not instantaneous. After 01 the decoder must wait: a 1 next means a longer word, and a 0 means s_2. The same bits read as 01 011 0 0111, which is $s_2 s_3 s_1 s_4$.

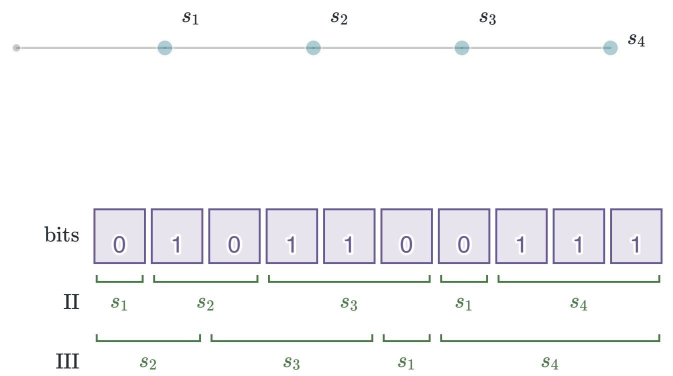


FIGURE 6.10 The three codes drawn as trees, shown at the last step. The tree of Code III is faint, and the bits 0101100111 are read with Codes II and III. In Code III each codeword lies on the path to the next.

The Kraft inequality

Codeword lengths can be tested before the codewords are chosen. A codeword of length l owns the fraction 2^{-l} of the unit interval: all long bit strings that start with it. In a prefix code these fractions do not overlap, so they add to at most one.

KRAFT INEQUALITY

$$\sum_{k=1}^{K} 2^{-l_k} \leq 1$$

A prefix code with lengths l_k exists exactly when the sum is at most one.

KRAFT SUMS OF THE THREE CODES

$$\begin{aligned} \mathrm{I}: \tfrac{1}{2}+\tfrac{1}{2}+\tfrac{1}{4}+\tfrac{1}{4} &= 1.5 \\ \mathrm{II}: \tfrac{1}{2}+\tfrac{1}{4}+\tfrac{1}{8}+\tfrac{1}{8} &= 1 \\ \mathrm{III}: \tfrac{1}{2}+\tfrac{1}{4}+\tfrac{1}{8}+\tfrac{1}{16} &= 0.9375 \end{aligned}$$

Code I gives $1.5 > 1$, so no prefix code has its lengths. Code II gives exactly 1. Lengths $1, 2, 2, 3$ give $\frac{1}{2} + \frac{1}{4} + \frac{1}{4} + \frac{1}{8} = 1.125 > 1$ and allow no prefix code either.

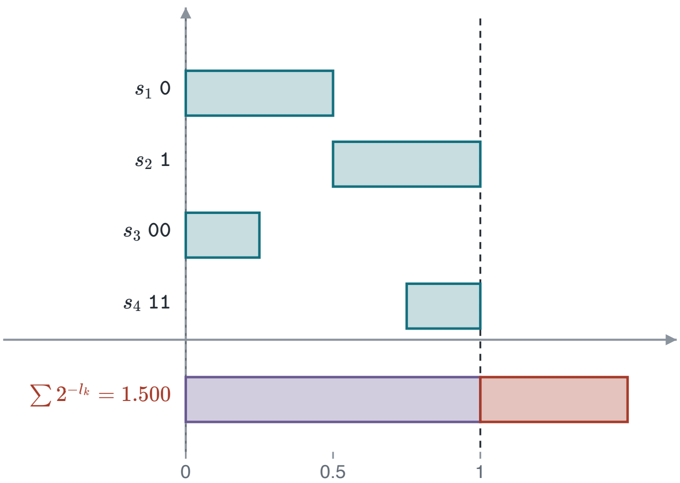


FIGURE 6.11 Each codeword of length l owns 2^{-l} of the unit interval, shown for Code I. Its stretches overlap, and the Kraft sum at the bottom is 1.5, past 1 (red).

⚠ LENGTHS, NOT CODEWORDS

Code III passes the test and is still not a prefix code. The inequality tests lengths only. The lengths $1, 2, 3, 4$ of Code III allow the prefix code 0, 10, 110, 1110.

The source-coding bound

The ideal length of a codeword is $-\log_2 p_k$, the information of its symbol. It is rarely a whole number, so round it up.

ROUNDED LENGTHS

$$l_k = \lceil -\log_2 p_k \rceil \implies -\log_2 p_k \le l_k < -\log_2 p_k + 1$$
$$2^{-l_k} \le p_k \implies \sum_k 2^{-l_k} \le \sum_k p_k = 1$$

The second line shows that the rounded lengths pass the Kraft test. So a prefix code with these lengths exists.

Multiply the first line by p_k and sum over k. The left side becomes $H(S)$, and the middle becomes $\bar{L}$.

SOURCE-CODING BOUND

$$H(S) \le \bar{L} < H(S) + 1$$

The rounding adds less than one bit a symbol. If every $p_k = 2^{-l_k}$, the source is **dyadic**. Then no length is rounded, and $\bar{L} = H(S)$.

The rounding bit can be spread over a block. Apply the bound to the extension S^n, whose entropy is $nH(S)$, and divide by n.

BLOCKS OF n SYMBOLS

$$nH(S) \le L_n < nH(S) + 1 \implies H(S) \le \frac{L_n}{n} < H(S) + \frac{1}{n}$$

Blocks of $n = 10$ symbols sit at most $1/10 = 0.1$ bit a symbol above H. The price is the codebook, which grows as K^n. For three symbols that is $3^{10} = 59\,049$ words.

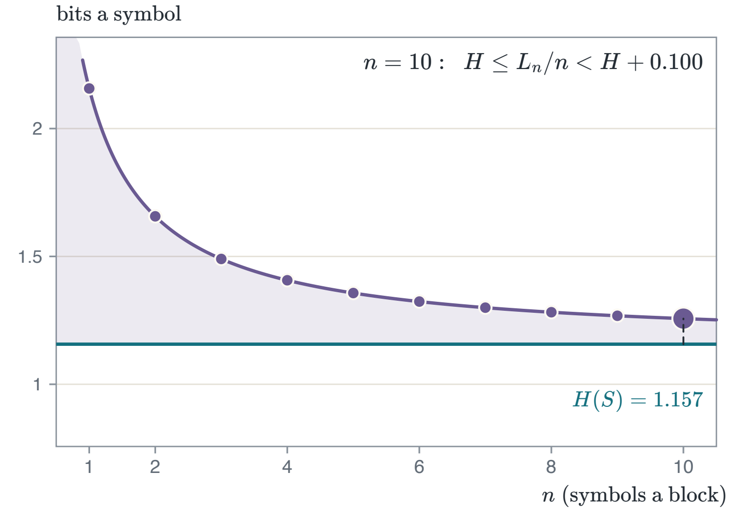


FIGURE 6.12 Bounds on the bits a symbol for blocks of n symbols from $0.7, 0.2, 0.1$, shown at $n = 10$. The gap of $1/n$ above $H(S) = 1.157$ closes as n grows.

Rate and distortion

When some error is allowed, fewer bits suffice. The **rate–distortion function** gives the fewest bits a sample for a mean-square error D.

For a Gaussian source of variance σ^2 the function has a closed form.

GAUSSIAN SOURCE

$$D(R) = \sigma^2\, 2^{-2R} \quad\Longleftrightarrow\quad R(D) = \tfrac{1}{2}\log_2 \frac{\sigma^2}{D}$$

One more bit multiplies D by $2^{-2(R+1)}/2^{-2R} = \frac{1}{4}$. In decibels that is $10\log_{10} 4 = 6.02$ dB a bit, the same rule as the quantizers of Chapter 1.

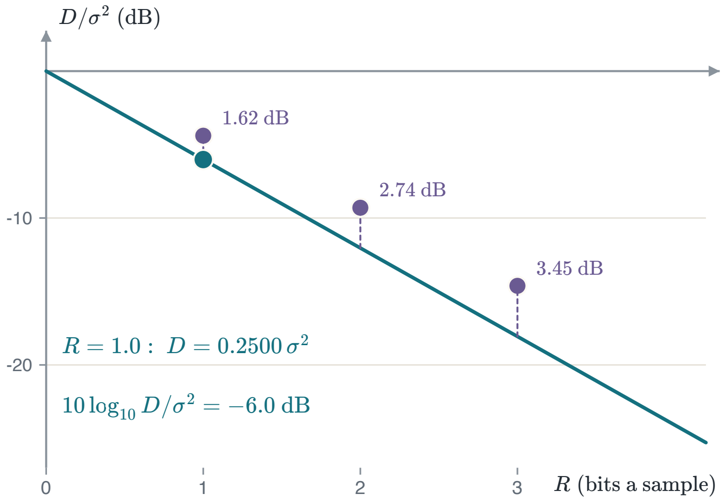


FIGURE 6.13 The least distortion of a Gaussian source at R bits a sample (line), and the Lloyd–Max quantizers of Chapter 1 (dots). Shown at $R = 1$, where $D = 0.25\,\sigma^2$.

⚠ QUANTIZERS

The Lloyd–Max quantizer of Chapter 1 codes one sample at a time. It sits 1.62 dB above $D(R)$ at 1 bit and 2.74 dB above it at 2 bits. Coding blocks of samples closes the gap.

Codes around us

Morse code and UTF-8 give frequent symbols short codewords. That is the idea of every variable-length code. Four everyday codes:

- Morse code gives common letters short signals: e is one dot. Each letter lasts $T = \sum_i t_i + (m-1)$ units, with a dot 1 and a dash 3.
- UTF-8 writes a character in 1 to 4 bytes: 1 for $b \le 7$, else $\lceil (b-1)/5 \rceil$. A lead byte never starts another, so it is a prefix code.
- Telephone country codes: $+1$, $+30$, $+351$, $+971$. No code is the start of another, so a switch knows where the code ends without a separator.
- Braille sets each letter in a cell of six dots, raised or flat: $2^6 = 64$ patterns, 6 bits a letter whatever its frequency.

PREFIX CODES

UTF-8 and the country codes need no separators. A reader knows where each codeword ends.

⚠ FIXED LENGTH

Braille spends 6 bits a letter. Without its spaces, the speech in Section 6.1 carries 4.04 bits a letter.

6.3 Huffman and Lempel–Ziv coding

Huffman coding

Huffman coding builds a prefix code by repeated merges.

ⓘ HUFFMAN ALGORITHM

1. Sort the probabilities.
2. Merge the two smallest into one entry whose probability is their sum. Label the pair 0 and 1, and sort again.
3. At one entry, read each codeword from the last merge back to its symbol.

Apply it to the source $0.4, 0.2, 0.2, 0.1, 0.1$ for $s_1, \ldots, s_5$. A merged entry that ties with others is placed above them.

THE MERGES

$$0.1 + 0.1 = 0.2, \qquad 0.2 + 0.2 = 0.4, \qquad 0.4 + 0.2 = 0.6, \qquad 0.6 + 0.4 = 1$$

CODEWORDS

$$\texttt{00},\ \texttt{10},\ \texttt{11},\ \texttt{010},\ \texttt{011}$$

Read back from the last merge, these are the codewords of s_1 to s_5. The two most likely symbols get two bits, the two least likely three.

The lengths are $2, 2, 2, 3, 3$. Weight each by its probability to get $\bar{L}$, then divide H by it.

AVERAGE LENGTH AND EFFICIENCY

$$\begin{aligned}\bar{L} &= 0.4(2) + 0.2(2) + 0.2(2) + 0.1(3) + 0.1(3) \\ &= 0.8 + 0.4 + 0.4 + 0.3 + 0.3 = 2.2 \text{ bits}\end{aligned}$$

$$\eta = \frac{H(S)}{\bar{L}} = \frac{2.1219}{2.2} = 0.9645$$

The average 2.2 lies above $H = 2.1219$. No prefix code for single symbols has a smaller average length.

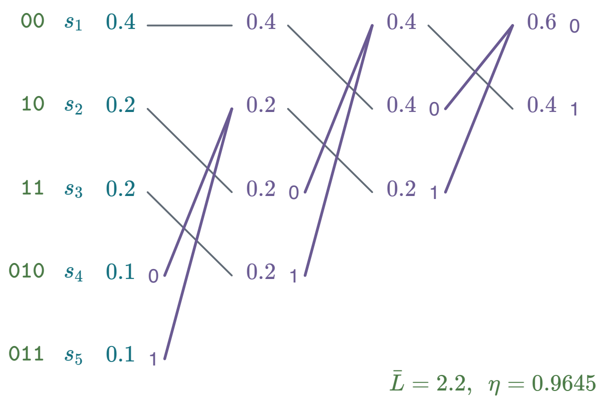


FIGURE 6.14 The Huffman merges for $0.4, 0.2, 0.2, 0.1, 0.1$. Each column merges the two smallest entries of the column before, labelled 0 and 1, and sorts again. Shown at the last step, with the codewords read back at the left.

Ties and variance

A merged entry often ties with other entries of the same probability. It may be placed above them or below them. Both choices give a Huffman code with the same average length.

> TWO HUFFMAN CODES FOR $0.4, 0.2, 0.2, 0.1, 0.1$
>
> $$\text{high: } l_k = 2, 2, 2, 3, 3, \qquad \bar{L} = 2.2$$
>
> $$\text{low: } l_k = 1, 2, 3, 4, 4, \qquad \bar{L} = 0.4 + 0.4 + 0.6 + 0.4 + 0.4 = 2.2$$
>
> Placed high, the codewords are $\texttt{00}, \texttt{10}, \texttt{11}, \texttt{010}, \texttt{011}$. Placed low, they are $\texttt{1}, \texttt{01}, \texttt{000}, \texttt{0010}, \texttt{0011}$.

The two codes differ in how far the lengths spread around $\bar{L}$. The **variance** of the codeword length measures that spread.

> VARIANCE OF THE CODEWORD LENGTH
>
> $$\sigma^2 = \sum_k p_k \left(l_k - \bar{L}\right)^2$$
>
> $$\begin{aligned} \sigma^2_{\text{high}} &= 0.8(0.2)^2 + 0.2(0.8)^2 = 0.032 + 0.128 = 0.16 \\ \sigma^2_{\text{low}} &= 0.4(1.2)^2 + 0.2(0.2)^2 + 0.2(0.8)^2 + 0.2(1.8)^2 \\ &= 0.576 + 0.008 + 0.128 + 0.648 = 1.36 \end{aligned}$$
>
> Placing the merged entry high gives the least variance. Lengths near $\bar{L}$ keep the bit rate steady, so a transmitter with a small buffer fills and empties it less.

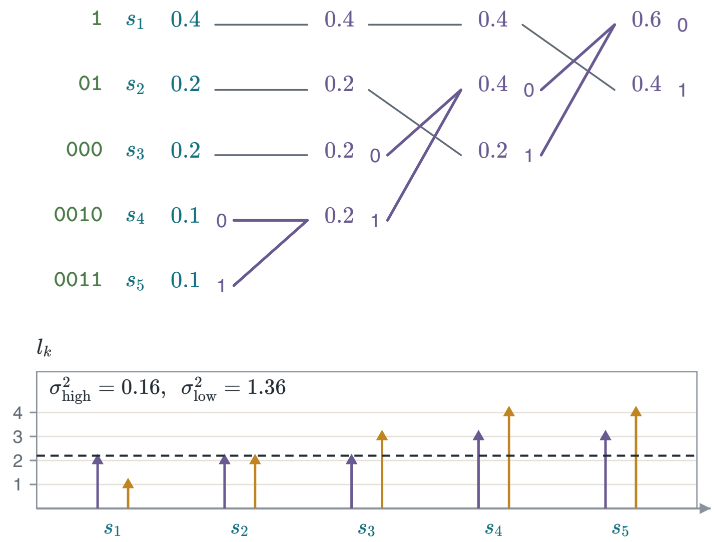


FIGURE 6.15 Top: the merges with ties placed low. Bottom: the codeword lengths of the two codes, ties high (violet) and ties low (amber), against $\bar{L} = 2.2$ (dashed).

Huffman codes for pairs of symbols

Huffman coding of the single symbols $0.7, 0.2, 0.1$ gives the codewords 0, 10, 11. Their average length is $0.7(1) + 0.2(2) + 0.1(2) = 1.3$ bits, against $H = 1.157$.

Coding blocks spreads the rounding loss. Build a Huffman code for the K^n blocks of n symbols, and divide its average length by n.

TABLE 6.3 Huffman codes for single symbols, pairs and triples of $0.7, 0.2, 0.1$, against $H = 1.1568$.

n	BLOCKS	$\bar{L}_n/n$ (BITS A SYMBOL)	BOUND $H + 1/n$	EFFICIENCY η
1	3	1.3	2.1568	0.8898
2	9	1.165	1.6568	0.9929
3	27	1.1753	1.4901	0.9842

Pairs cost $\bar{L}_2 = 2.33$ bits a pair, or $2.33/2 = 1.165$ bits a symbol. Triples cost 1.1753 bits a symbol, a little more than pairs.

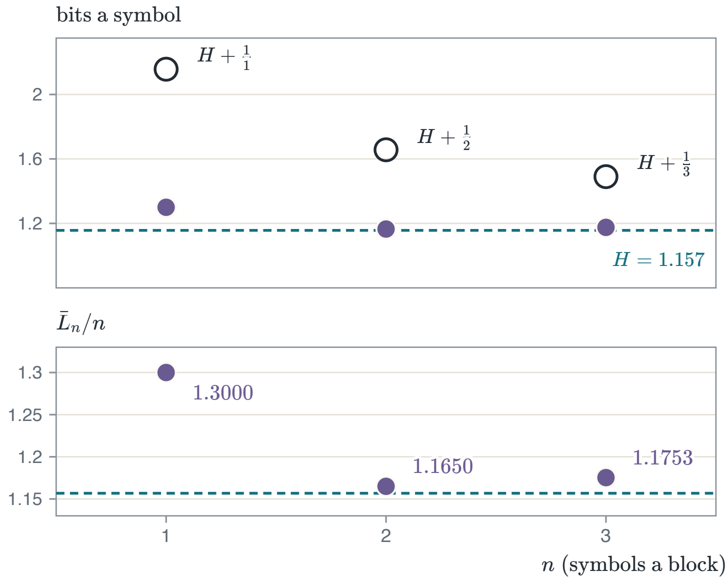


FIGURE 6.16 Huffman codes for single symbols, pairs and triples of $0.7, 0.2, 0.1$. Top: the bound $H + 1/n$ (rings) and the code (dots). Bottom: the code, enlarged.

> ⊗ COMMON ERROR
>
> Expect the bound $H + 1/n$ to fall with n, not the cost of every block code. Triples give 1.1753 bits a symbol, worse than the 1.165 of pairs.

Arithmetic coding

Arithmetic coding codes a whole message at once. It starts with the interval $[0, 1)$. Each symbol keeps the part of the current interval that its probability owns.

Code the message $s_1 s_1 s_2$ from the source $0.7, 0.2, 0.1$. The symbols own the first 70%, the next 20% and the last 10% of each interval.

> THE INTERVAL AFTER EACH SYMBOL
>
> $$\begin{aligned} s_1 :&\ [0, 0.7) \\ s_1 s_1 :&\ [0, 0.49) \\ s_1 s_1 s_2 :&\ [0.343, 0.441) \end{aligned}$$
>
> The last symbol keeps the part from 70% to 90% of $[0, 0.49)$: $0.7 \times 0.49 = 0.343$ and $0.9 \times 0.49 = 0.441$
> .

Each symbol scales the interval by its probability. So the width is the probability of the whole message.

> WIDTH
>
> $$w = \prod_i p(s_i) = 0.7 \times 0.7 \times 0.2 = 0.098$$

A binary fraction of l bits inside the interval names it. About $-\log_2 w$ bits are needed, plus one.

TAG

$$l = \lceil -\log_2 w \rceil + 1 = \lceil 3.35 \rceil + 1 = 5 \text{ bits}$$

The tag is $0.01100_2 = 0.375$. Every binary fraction that starts with 0.01100 lies in $[0.375, 0.40625)$, inside $[0.343, 0.441)$. The extra bit is paid once a message.

The tag length follows the width. When the width halves, the tag needs one more bit, because $-\log_2(w/2) = -\log_2 w + 1$.

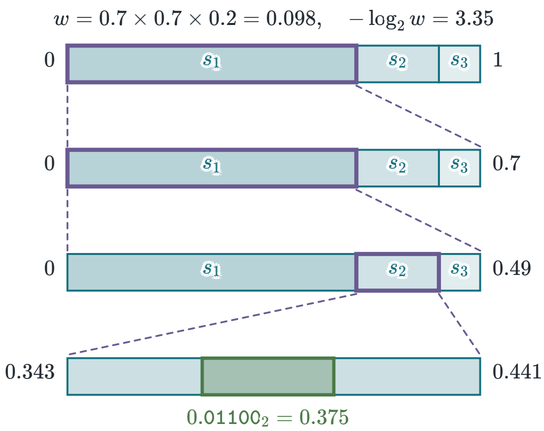


FIGURE 6.17 Arithmetic coding of $s_1 s_1 s_2$ with $0.7, 0.2, 0.1$. Each symbol keeps its share of the current interval. Shown at the last step, where the binary fraction 0.375 names the final interval.

Lempel–Ziv coding

Lempel–Ziv coding builds a dictionary of phrases from the stream itself.

ⓘ LEMPEL–ZIV PARSING

Read the stream. Cut off the shortest string not yet in the dictionary, and add it as a new entry. Each new phrase is an earlier phrase plus one bit.

Each phrase is sent as a pair. The pointer names the earlier phrase, and one more bit carries the new bit. The decoder builds the same dictionary from the pairs.

PHRASE CODE

$$(\text{pointer, new bit}): \quad \texttt{010} \to (4, \texttt{0})$$

Parse the 18-bit stream 000101110010100101. The dictionary starts with the empty phrase as entry 0. The table lists the eight phrases and the pair sent for each.

TABLE 6.4 Lempel–Ziv parsing of the stream 000101110010100101.

ENTRY i	PHRASE	PAIR SENT	BITS $\lceil \log_2 i \rceil + 1$
1	0	$(0, 0)$	1
2	00	$(1, 0)$	2
3	1	$(0, 1)$	3
4	01	$(1, 1)$	3
5	11	$(3, 1)$	4
6	001	$(2, 1)$	4
7	010	$(4, 0)$	4
8	0101	$(7, 1)$	4

Phrase 7, 010, is phrase 4 plus a 0, so it is sent as $(4, 0)$. The stream then goes on 0101. That is entry 7 plus a 1, a new phrase, sent as $(7, 1)$.

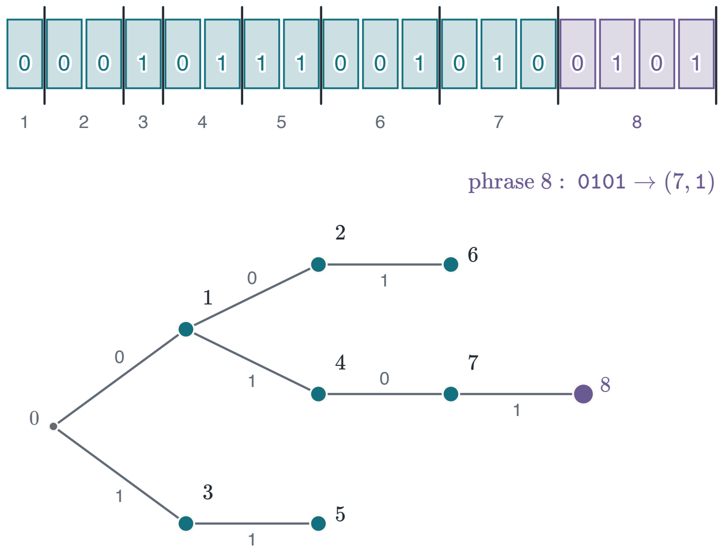


FIGURE 6.18 The 18-bit stream parsed into phrases, shown after the eighth phrase. Each new phrase is an earlier phrase plus one bit, so the dictionary grows as a tree.

Phrase i can point to any of the i entries 0 to $i-1$, so its pointer needs $\lceil \log_2 i \rceil$ bits. The last column of the table adds to $1+2+3+3+4+4+4+4 = 25$ bits.

> ⚠ SHORT STREAMS
>
> With a 3-bit pointer and one new bit, eight phrases take 32 bits for 18. A pointer of $\lceil \log_2 i \rceil$ bits for phrase i takes 25.

Lempel–Ziv on a long stream

Lempel–Ziv is a **universal** code: it is told nothing about the source, and it learns the frequent strings from the stream itself. On a long stream the phrases grow long, and each pointer stands for many source bits.

For c phrases the cost is a sum over the phrases.

COST OF c PHRASES

$$L = \sum_{i=1}^{c} \left(\lceil \log_2 i \rceil + 1 \right) \text{ bits}$$

Take a binary source with $P(1) = 0.1$, so $H_b(0.1) = 0.469$ bit a symbol. On one random stream the cost is 0.689 bits a source bit after 10^3 bits and 0.557 after 10^6.

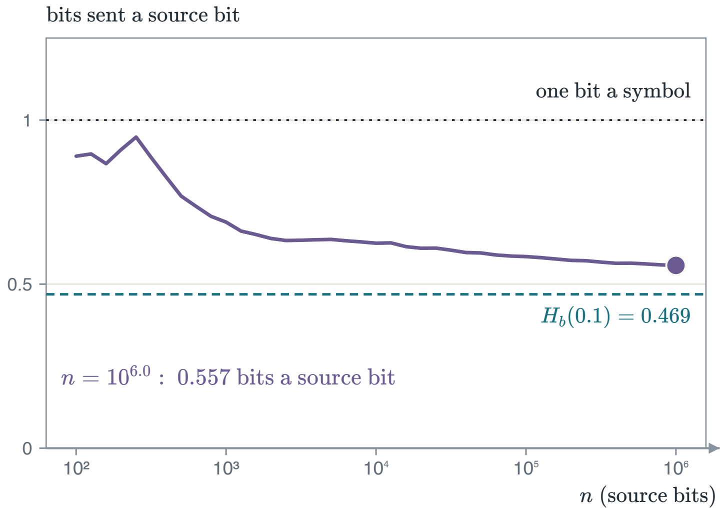


FIGURE 6.19 Lempel–Ziv on one random stream from a binary source with $P(1) = 0.1$: the bits sent a source bit against the stream length, and $H_b(0.1)$ (dashed). Shown at $n = 10^6$.

⚠ SLOW APPROACH

The cost falls toward H only as the phrases grow long, and it never falls below H. Any code for single binary symbols costs 1 bit a symbol here.

Compression around us

Each format below first turns the data into something with low entropy: copies, differences, zeros or runs.

- ZIP and gzip use DEFLATE: LZ77 copies, then Huffman codes. $\ell(i)$ is the longest earlier copy of the text from character i.
- PNG filters each row first: $d[n] = x[n] - x[n-1]$ takes 2.11 bits a pixel against 5.00. DEFLATE then codes $d[n]$.
- JPEG takes the DCT of each 8×8 block and quantizes it. Here 9 of the 64 values c_k are nonzero. The runs of zeros are Huffman-coded.
- A fax line of $\sum_i \ell_i = 1728$ pixels is sent as run lengths ℓ_i, white and black in turn, each Huffman-coded.

THEN HUFFMAN

A Huffman code, or a close cousin, then spends bits by frequency.

⚠ LOSSY AND LOSSLESS

ZIP, PNG and fax lose nothing. JPEG throws information away in its quantizer before any coding.

6.4 Channels and mutual information

The discrete memoryless channel

A **discrete memoryless channel** takes an input symbol x_j and returns an output symbol y_k with probability $p(y_k \mid x_j)$. Memoryless means each use ignores the others.

The transition probabilities form the **channel matrix**, with one row for each input.

CHANNEL MATRIX

$$\mathbf{P} = \begin{bmatrix} p(y_0 \mid x_0) & p(y_1 \mid x_0) \\ p(y_0 \mid x_1) & p(y_1 \mid x_1) \end{bmatrix} = \begin{bmatrix} 0.8 & 0.2 \\ 0.3 & 0.7 \end{bmatrix}$$

Row j is the output distribution of input x_j, so it sums to one: $0.8 + 0.2 = 0.3 + 0.7 = 1$. The columns need not sum to one. Here they give 1.1 and 0.9.

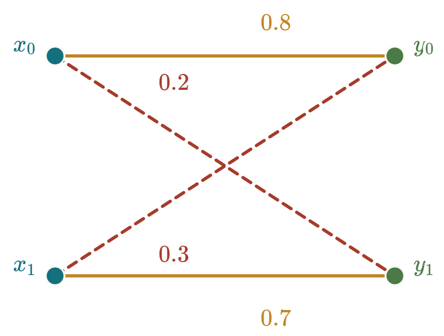


FIGURE 6.20 The channel above as a diagram. Each input has a line to each output it can reach, labelled with the probability of that move. Amber lines keep the symbol, and red lines change it.

Input and output distributions

The channel fixes $p(y \mid x)$. The transmitter chooses the **input distribution** $p(x)$. Together they give the joint and the output distributions.

JOINT AND OUTPUT DISTRIBUTIONS

$$p(x_j, y_k) = p(y_k \mid x_j)\, p(x_j)$$

$$p(y_k) = \sum_j p(y_k \mid x_j)\, p(x_j)$$

The output probability adds the joint probabilities of every path that ends at y_k.

With $p(x_0) = 0.75$, the output y_0 comes from x_0 with probability 0.6 and from x_1 with probability 0.075. Add the two paths.

OUTPUT DISTRIBUTION FOR $p(x_0) = 0.75$

$$\begin{aligned} p(y_0) &= 0.8 \times 0.75 + 0.3 \times 0.25 \\ &= 0.6 + 0.075 = 0.675 \\ p(y_1) &= 1 - 0.675 = 0.325 \end{aligned}$$

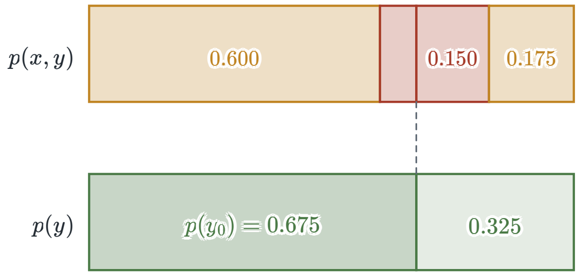


FIGURE 6.21 The four joint probabilities of the channel $0.8/0.2$, $0.3/0.7$, drawn to width: kept (amber) and moved (red). Below, the output distribution they add to (green), shown at $p(x_0) = 0.75$.

The binary symmetric channel

The **binary symmetric channel** (BSC) has two inputs and two outputs. Each bit arrives flipped with probability p, the **crossover probability**, and intact with probability $1 - p$, whatever its value.

BINARY SYMMETRIC CHANNEL

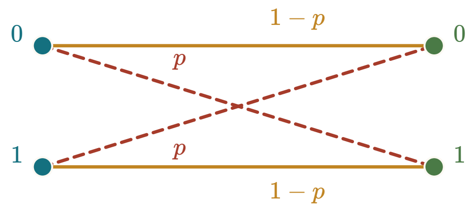
$$\mathbf{P} = \begin{bmatrix} 1-p & p \\ p & 1-p \end{bmatrix}$$

The matched-filter receiver of Chapter 4, deciding each BPSK bit of Chapter 5, hands on a BSC. Its crossover is the BPSK bit error. Here $Q(x) = \frac{1}{2}\,\mathrm{erfc}(x/\sqrt{2})$ is the Gaussian tail.

BSC FROM HARD-DECISION BPSK

$$p = Q\left(\sqrt{2E_b/N_0}\right)$$

$$\begin{aligned} E_b/N_0 &= 4\ \mathrm{dB} = 10^{0.4} = 2.512 \\ p &= Q\left(\sqrt{2 \times 2.512}\right) = Q(2.241) = 0.0125 \end{aligned}$$



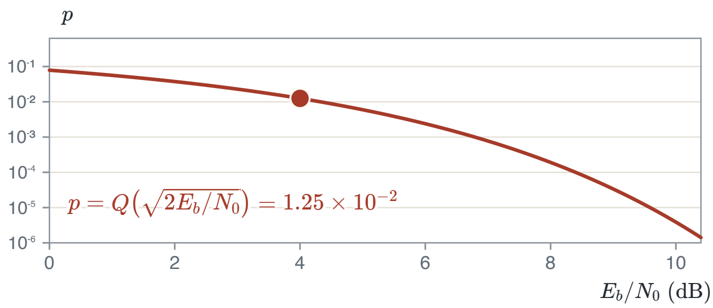


FIGURE 6.22 Top: the binary symmetric channel. Bottom: its crossover when BPSK is decided bit by bit, shown at $E_b/N_0 = 4$ dB, where $p = 0.0125$.

⚠ HARD DECISIONS

Deciding each bit throws away how sure the receiver was. The gallery at the end of Section 6.5 plots the capacity left after that decision.

Joint entropy and the chain rule

The **joint entropy** of two variables is the entropy of the pair, treated as one symbol.

JOINT ENTROPY

$$H(X,Y) = -\sum_{j,k} p(x_j, y_k) \log_2 p(x_j, y_k)$$

The uncertainty about Y left once X is known is the conditional entropy $H(Y \mid X) = -\sum_{j,k} p(x_j, y_k) \log_2 p(y_k \mid x_j)$. The joint probability splits as $p(x,y) = p(x)\,p(y \mid x)$. Its logarithm is a sum, and averaging each term gives the **chain rule**.

CHAIN RULE

$$\begin{aligned} H(X,Y) &= -\sum_{j,k} p(x_j, y_k)\left[\log_2 p(x_j) + \log_2 p(y_k \mid x_j)\right] \\ &= H(X) + H(Y \mid X) \end{aligned}$$

$$H(X,Y) = H(X) + H(Y \mid X) = H(Y) + H(X \mid Y)$$

Learn X first, then what is left of Y. The second form follows the same way with the roles of X and Y swapped.

Take the joint pmf $p(x_0, y_0) = p(x_1, y_1) = 0.4$ and $p(x_0, y_1) = p(x_1, y_0) = 0.1$.

CHAIN RULE FOR ONE PMF

$$H(X,Y) = -2(0.4)\log_2 0.4 - 2(0.1)\log_2 0.1 = 1.7219$$

$$H(X) = 1, \qquad H(Y \mid X) = H_b(0.2) = 0.7219$$

Each input has probability 0.5, so $H(X) = 1$. Given either input, Y agrees with probability $0.4/0.5 = 0.8$, so $H(Y \mid X) = H_b(0.2)$. The two parts add to 1.7219, as the chain rule says.

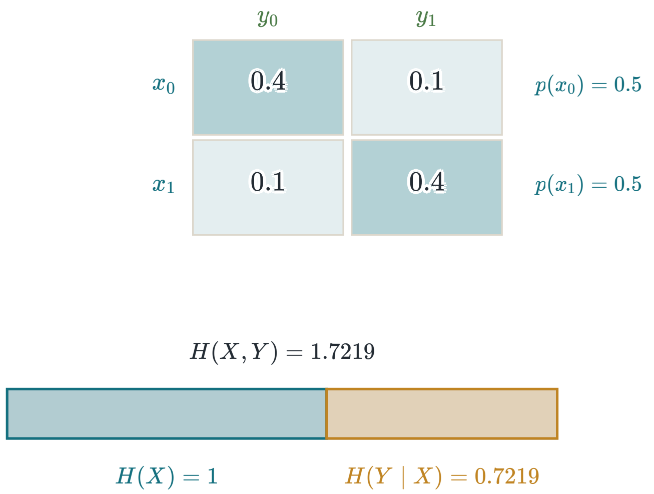


FIGURE 6.23 The joint pmf as tiles, and its joint entropy as a bar, shown at the last step: $H(X)$ fills part of the bar, and $H(Y \mid X)$ fills the rest.

Conditional entropy

The receiver sees Y and wants X. The **conditional entropy** $H(X \mid Y)$ is the uncertainty about the input left after the output is seen, averaged over the outputs.

CONDITIONAL ENTROPY

$$H(X \mid Y) = \sum_k p(y_k)\, H(X \mid Y = y_k)$$

Return to the channel $0.8/0.2$, $0.3/0.7$ with $p(x_0) = 0.75$. Bayes' rule gives the input probabilities after each output.

AFTER EACH OUTPUT

$$P(x_0 \mid y_0) = \frac{0.8 \times 0.75}{0.675} = 0.889, \qquad H_b(0.889) = 0.503$$

$$P(x_0 \mid y_1) = \frac{0.2 \times 0.75}{0.325} = 0.462, \qquad H_b(0.462) = 0.996$$

$$H(X \mid Y) = 0.675(0.503) + 0.325(0.996) = 0.663$$

Before the output, $H(X) = H_b(0.75) = 0.811$ bit. Output y_0 removes much of the doubt. Output y_1 leaves the input almost a coin toss.

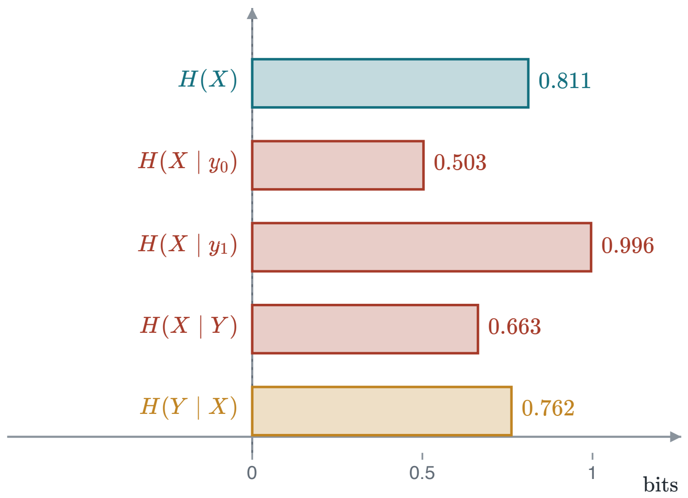


FIGURE 6.24 The channel $0.8/0.2$, $0.3/0.7$ with $p(x_0) = 0.75$: the uncertainty about X before the output, after each output, their average, and the other conditional entropy $H(Y \mid X)$.

COMMON ERROR

Keep $H(X \mid Y)$ apart from $H(Y \mid X)$. Here $H(Y \mid X) = 0.75H_b(0.2) + 0.25H_b(0.3) = 0.762$, but $H(X \mid Y) = 0.663$.

In a BSC with equally likely inputs, either output leaves the input wrong with probability p. So $H(X \mid Y) = H_b(p)$. At $p = 0.1$ that is $H_b(0.1) = 0.469$ bit.

Mutual information

Mutual information is the uncertainty about the input that the output removes. It is the uncertainty before the output, less the uncertainty after it.

MUTUAL INFORMATION

$$I(X;Y) = H(X) - H(X \mid Y)$$

For a BSC with equal inputs, $H(X) = 1$ and $H(X \mid Y) = H_b(p)$. Substitute both.

BSC WITH EQUAL INPUTS

$$I(X;Y) = 1 - H_b(p)$$

At $p = 0$ the full bit gets through. At $p = \frac{1}{2}$ the output is a coin toss and $I = 0$. At $p = 0.1$, $I = 1 - 0.469 = 0.531$ bit a use.

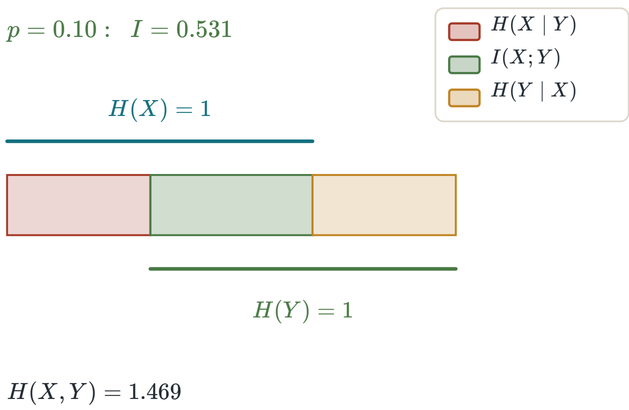


FIGURE 6.25 A BSC with equal inputs, shown at $p = 0.1$. The bar is $H(X,Y)$, the line above it $H(X)$ and the line below it $H(Y)$. Their overlap is $I(X;Y)$.

Properties of mutual information

Mutual information is symmetric. It follows from Bayes' rule: $p(x,y)$ is the same read either way. The chain rule gives $H(X \mid Y) = H(X,Y) - H(Y)$ and $H(Y \mid X) = H(X,Y) - H(X)$. Substitute either one into the definition.

SYMMETRY AND THE JOINT ENTROPY

$$I(X;Y) = H(X) - H(X \mid Y) = H(Y) - H(Y \mid X) = I(Y;X)$$

$$I(X;Y) = H(X) + H(Y) - H(X,Y) \geq 0$$

Adding $H(X)$ and $H(Y)$ counts the shared part twice, so the joint entropy is subtracted once.

A BSC with $p = 0.25$ and equal inputs has $H(X) = H(Y) = 1$ and $H(X,Y) = 1.811$. So $I(X;Y) = 1 + 1 - 1.811 = 0.189$ bit.

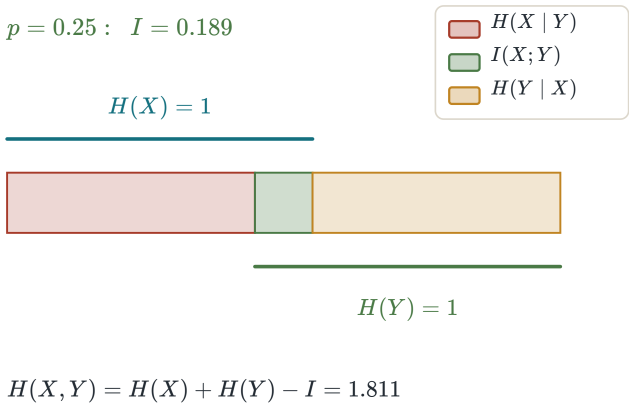


FIGURE 6.26 The bars for a BSC with $p = 0.25$ and equal inputs. The shared middle is the same whether it is read from X or from Y.

> NEVER NEGATIVE
>
> Seeing Y never adds to the uncertainty about X on average: $H(X \mid Y) \le H(X)$. Equality holds when X and Y are independent. Then nothing is shared, and $I(X;Y) = 0$.

Information around us

In each case below, $I = H(X) - H(X \mid Y)$ is how much the observation lowers the uncertainty.

- A forecast: rain 30% of days, forecast on 80% of rainy days and 10% of dry ones. $I(W;F) = 0.348$ of $H(W) = 0.881$ bits.
- A test with 99% sensitivity and a 5% false-positive rate. For a condition in 1% of people, $I(D;T) = 0.041$ bit a test.
- A BPSK receiver that decides each bit. With equal bits $I = 1 - H_b(p)$ and $p = Q\left(\sqrt{2E/N_0}\right)$: 0.903 bit at 4 dB.
- An 8-level quantizer of step 0.5σ on a Gaussian sample. Its output is fixed by the input, so $I(X;Q) = H(Q) = 2.888$ bits.

> ACCURATE IS NOT INFORMATIVE
>
> A good test of a rare condition says little on average, because it almost always reads negative.

> DETERMINISTIC OUTPUT
>
> When Y is a function of X, $H(Y \mid X) = 0$ and $I(X;Y) = H(Y)$.

6.5 Channel capacity

Channel capacity

Mutual information depends on the channel and on the input distribution. The channel fixes $p(y \mid x)$, and the transmitter picks the input distribution. The choice that lets the most through gives the **capacity**.

CHANNEL CAPACITY

$$C = \max_{p(x)} I(X;Y) \quad \text{bits per use}$$

$I(X;Y)$ depends on the input. C does not: the maximum has removed it, so C is a property of the channel alone.

The BSC is symmetric in its two inputs. So $I(X;Y)$ is symmetric about $q = p(x_0) = \frac{1}{2}$, and its peak sits there, with $C = 0.531$ at $p = 0.1$. The Z-channel of Example 6.2 is not symmetric. Its peak sits at $q = 0.6$, with $C = 0.322$.

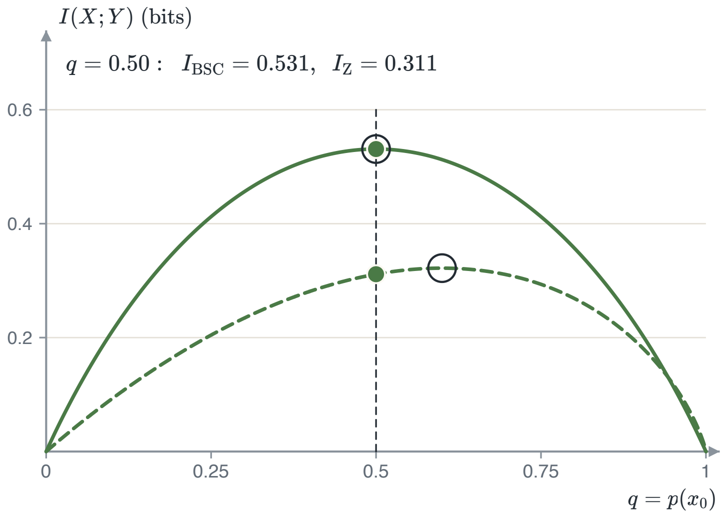


FIGURE 6.27 Mutual information against the input distribution, for a BSC with $p = 0.1$ (solid) and the Z-channel (dashed), shown at $q = \frac{1}{2}$. The ring on each peak is its capacity.

Capacity of the binary symmetric channel

Write the mutual information from the output side and bound $H(Y)$.

CAPACITY OF THE BSC

$$\begin{aligned} I(X;Y) &= H(Y) - H(Y \mid X) \\ &= H(Y) - H_b(p) \\ &\leq 1 - H_b(p) \end{aligned}$$

$$C = 1 - H_b(p)$$

Whatever the input, each output is wrong with probability p, so $H(Y \mid X) = H_b(p)$. A binary output has $H(Y) \leq 1$. Equal inputs give equal outputs and make $H(Y) = 1$.

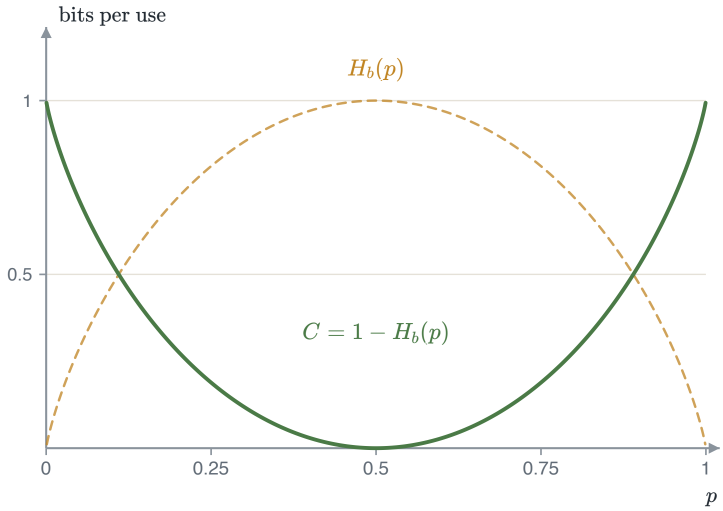


FIGURE 6.28 The capacity $C = 1 - H_b(p)$ of the binary symmetric channel (solid) against its crossover p, over $H_b(p)$ (dashed). It is 0 at $p = \frac{1}{2}$.

The capacity is 0 at $p = \frac{1}{2}$, where the output ignores the input. At $p = 1$ every bit flips, the receiver flips it back, and $C = 1$. At $p = 0.11$, $H_b = 0.500$ and $C = 0.500$ bit a use.

The binary erasure channel

The **binary erasure channel** (BEC) never flips a bit. Each bit arrives intact with probability $1 - \epsilon$, or as a flagged erasure e with probability ϵ.

An intact bit leaves no doubt about the input. An erasure leaves the input as uncertain as before. So $H(X \mid Y) = \epsilon H(X)$, and the mutual information follows.

CAPACITY OF THE BEC

$$I(X;Y) = H(X) - \epsilon H(X) = (1 - \epsilon)H(X)$$

$$C = 1 - \epsilon$$

Equal inputs give $H(X) = 1$ and reach the capacity. The bits that arrive are certain, and only the erased fraction is lost. At $\epsilon = 0.1$ the BEC carries 0.9 bit a use, and a BSC with $p = 0.1$ carries 0.531.

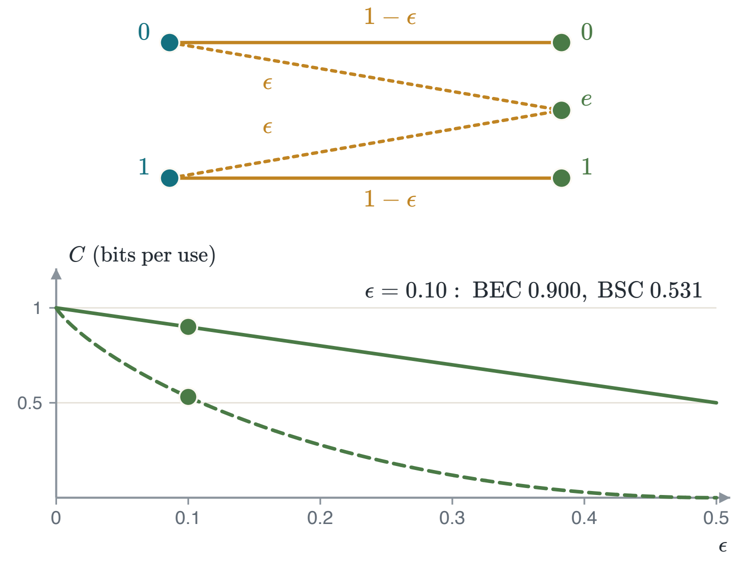


FIGURE 6.29 Top: the binary erasure channel, where a lost bit arrives as e. Bottom: its capacity (solid) against the BSC with $p = \epsilon$ (dashed), shown at $\epsilon = 0.1$.

> ⚠ FLAGGED AGAINST HIDDEN
>
> The BSC hides its errors among good bits, so it loses $H_b(\epsilon)$ bits a use. That is more than ϵ for $\epsilon < \frac{1}{2}$
> .

✎ EXAMPLE 6.2 — THE Z-CHANNEL

GIVEN A 0 is always received as 0. A 1 is received as 0 or 1 with probability $\frac{1}{2}$ each. Let $q = P(X = 0)$.

FIND The capacity C and the input distribution that reaches it.

METHOD The channel is not symmetric, so equal inputs need not be best. Write $I(X;Y) = H(Y) - H(Y \mid X)$ as a function of q, and set its derivative to zero.

SOLUTION Only $X = 1$ leaves the output uncertain, with $H_b(\frac{1}{2}) = 1$ bit, so $H(Y \mid X) = 1 - q$. The output is 1 with probability $(1-q)/2$, so $I(q) = H_b\left(\frac{1-q}{2}\right) - (1-q)$. The derivative of $H_b(x)$ is $\log_2 \frac{1-x}{x}$, and $x = \frac{1-q}{2}$ has $\mathrm{d}x/\mathrm{d}q = -\frac{1}{2}$. So $\mathrm{d}I/\mathrm{d}q = 1 - \frac{1}{2}\log_2 \frac{1+q}{1-q}$. Setting it to zero gives $\frac{1+q}{1-q} = 4$, so $q^* = 0.6$. Then $x = 0.2$ and $C = H_b(0.2) - 0.4 = 0.7219 - 0.4 = 0.3219$ bit a use.

CHECK Write $H_b(0.2) = 0.2\log_2 5 + 0.8\log_2 \frac{5}{4} = \log_2 5 - 1.6$. Then $C = \log_2 5 - 2 = \log_2 \frac{5}{4} = 0.3219$, the same value. The best input sends the reliable 0 three times in five.

> ⊗ COMMON ERROR
>
> Use the maximum over q, not equal inputs. At $q = \frac{1}{2}$, $I = H_b(0.25) - 0.5 = 0.3113$, below C. On an asymmetric channel the best input is not uniform.

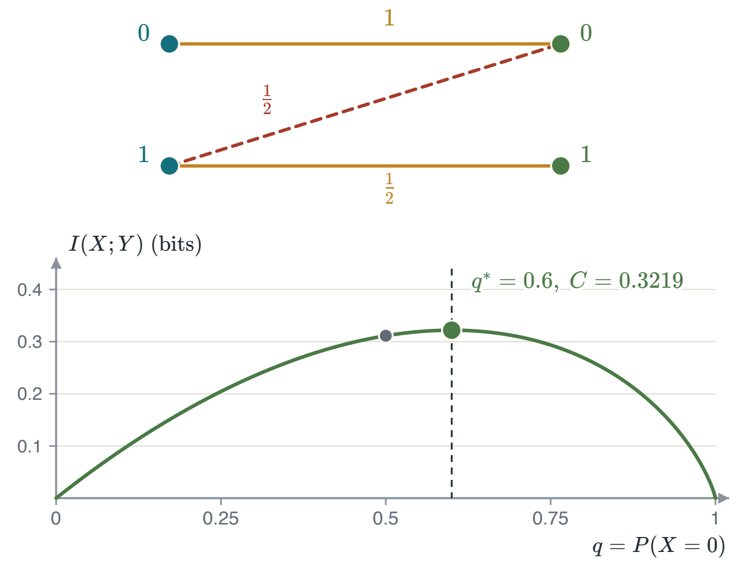


FIGURE 6.30 Top: the Z-channel. A 0 always arrives, and a 1 turns into 0 half the time. Bottom: $I(X;Y)$ against q, with its peak at $q^* = 0.6$ and the value at $q = \frac{1}{2}$ (grey).

EXAMPLE 6.3 — A SYMMETRIC THREE-OUTPUT CHANNEL

GIVEN Three inputs and three outputs. Each row of the channel matrix is a shift of $0.6, 0.2, 0.2$, so each column is too.

FIND The capacity, and the input distribution that reaches it.

METHOD Every row has the same entropy, so $H(Y \mid X) = H(0.6, 0.2, 0.2)$ for any input. Inputs of $\frac{1}{3}$ each make the outputs equal, because every column holds $0.6, 0.2, 0.2$. Then $H(Y) = \log_2 K$, its largest value, and $C = \log_2 K - H(\text{row})$.

SOLUTION $H(0.6, 0.2, 0.2) = 0.442 + 0.464 + 0.464 = 1.371$ bits. So $C = \log_2 3 - 1.371 = 1.585 - 1.371 = 0.2140$ bit a use.

CHECK One bit needs at least $1/0.214 = 4.67$ uses of this channel. The uniform input reaches C, as the symmetry of the rows requires.

COMMON ERROR

Use $\log_2 3 - H(\text{row})$, not $\log_2 3$, for the capacity. The value $\log_2 3 = 1.585$ is the most any three-output channel carries.

	y_0	y_1	y_2
x_0	0.6	0.2	0.2
x_1	0.2	0.6	0.2
x_2	0.2	0.2	0.6
$p(y)$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{3}$

each row: $H = 1.371$

FIGURE 6.31 The channel matrix of Example 6.3 as tiles. Each row is a shift of $0.6, 0.2, 0.2$. Equal inputs give equal outputs, shown in the bottom row.

Reliable transmission over a noisy channel

Capacity also counts the messages a channel keeps apart. Take a channel with four inputs, where each input reaches two of four outputs with probability $\frac{1}{2}$ each. Every output can then come from two inputs.

Use only x_0 and x_2, whose outputs do not overlap. The receiver never confuses them, so the channel carries one bit a use with no errors.

	y_0	y_1	y_2	y_3
x_0	$\frac{1}{2}$	$\frac{1}{2}$		
x_1				
x_2			$\frac{1}{2}$	$\frac{1}{2}$
x_3				

use x_0, x_2 only: no confusion, 1 bit a use

FIGURE 6.32 A four-input channel whose outputs overlap, as tiles. Using only x_0 and x_2 leaves no output that can come from both.

Long blocks do the same for a BSC. A word of n bits almost surely lands among about $2^{nH_b(p)}$ likely outputs, its cloud. Codewords whose clouds do not overlap can be told apart.

> COUNTING CODEWORDS
>
> $$M \approx \frac{2^n}{2^{nH_b(p)}} = 2^{n(1-H_b(p))} = 2^{nC}$$
>
> There are 2^n output words in all. So about 2^{nC} codewords fit, which is C bits a use. At $n = 100$ and $p = 0.11$, $C = 0.5$ and about $2^{100\times0.5} = 2^{50}$ codewords fit.

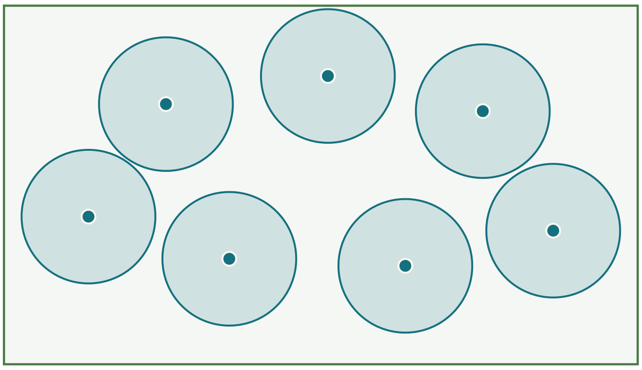


FIGURE 6.33 Long blocks of a BSC: each codeword owns a cloud of about $2^{nH_b(p)}$ likely outputs among the 2^n output words.

Repetition codes

The simplest channel code sends each bit n times, with n odd, and decides by majority. It is a **repetition code** of rate $R = 1/n$. The vote fails when more than half the copies flip.

ERROR OF A REPETITION CODE

$$P_e = \sum_{k>n/2} \binom{n}{k} p^k (1-p)^{n-k}$$

On a BSC with $p = 0.1$, one copy fails with $P_e = 0.1$. Three and five copies need two and three flips to fail.

REPETITION ON A BSC WITH $p = 0.1$

$$\begin{aligned} n = 3: \quad P_e &= 3p^2(1-p) + p^3 \\ &= 0.027 + 0.001 = 0.028 \\ n = 5: \quad P_e &= 10p^3(1-p)^2 + 5p^4(1-p) + p^5 \\ &= 0.0081 + 0.00045 + 0.00001 \approx 0.0086 \end{aligned}$$

At $n = 15$, $P_e = 3.36 \times 10^{-5}$, but the rate has fallen to $1/15$.

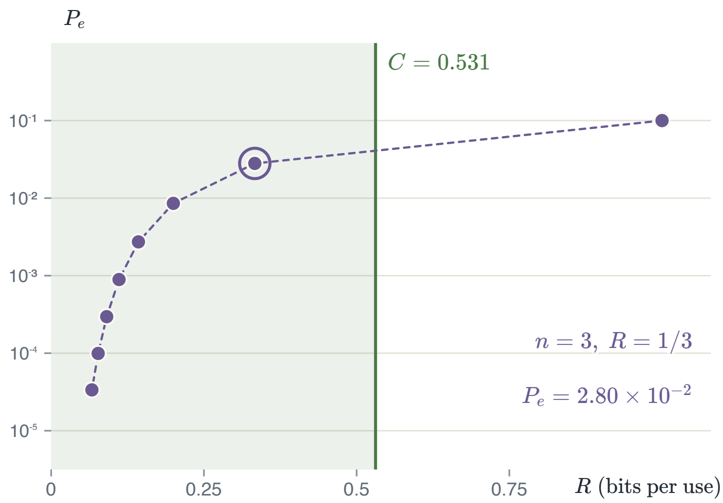


FIGURE 6.34 Repetition codes on a BSC with $p = 0.1$: the error after a majority vote against the rate $1/n$, for $n = 1, 3, \ldots, 15$, with $n = 3$ ringed. The green band marks the rates below C.

> ⚠ RATE FALLS TOO
>
> Repetition drives the error to zero only as the rate $1/n$ goes to zero. The channel coding theorem promises far better: any rate below $C = 0.531$.

The channel coding theorem

The **channel coding theorem** states which rates can be made reliable. Here R is in bits a channel use.

> CHANNEL CODING THEOREM
>
> $$R < C: \ P_e \to 0 \text{ is possible}, \qquad R > C: \ \text{it is not}$$
>
> For $R < C$, codes exist with rate close to C and error as small as asked. They need long blocks, and the theorem does not build them. For $R > C$, no code of any length is reliable.

A code of rate $R = 0.5$ over a BSC needs $1 - H_b(p) > 0.5$. That holds for $p < 0.11$, where $H_b = 0.5$. The repetition codes sit far below the limit, because their rate falls with their error. The theorem says that is unnecessary.

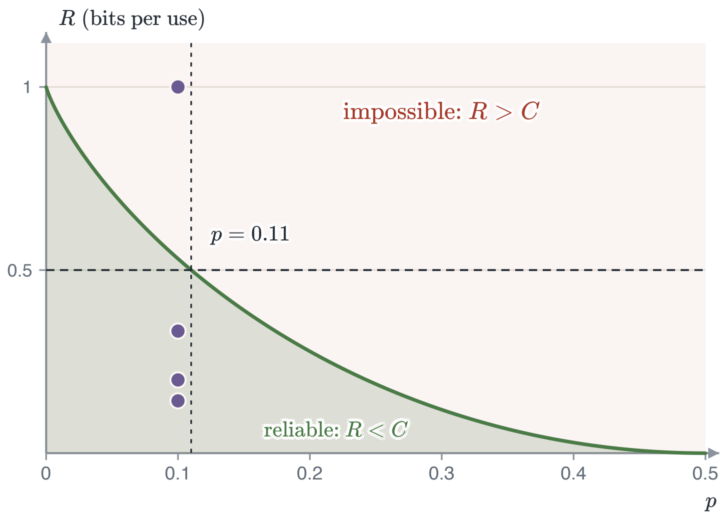


FIGURE 6.35 Over a BSC: rates below $C(p) = 1 - H_b(p)$ can be made reliable, and rates above it cannot. The violet dots are the repetition codes at $p = 0.1$.

A glimpse of codes: the Hamming code

Real codes add structure. A single **parity bit** makes the number of ones in a word even. One error makes it odd, so the error is detected, but its position is not known.

The $(7, 4)$ **Hamming code** sends four data bits and three parity bits, a rate of $4/7$. Each parity bit keeps one group of bits even, drawn as one of three circles. Its 16 codewords differ pairwise in at least $d_{\min} = 3$ places.

ERRORS CORRECTED

$$t = \left\lfloor \frac{d_{\min} - 1}{2} \right\rfloor = \left\lfloor \frac{3 - 1}{2} \right\rfloor = 1$$

A word with one error is still nearer its own codeword than any other. So one error can be corrected.

The bits sit at positions 1 to 7. Check s_1 covers the positions with a 1 in the units place of their binary index: $1, 3, 5, 7$. Check s_2 covers $2, 3, 6, 7$, and check s_4 covers $4, 5, 6, 7$. The failing checks form the **syndrome** $s_4 s_2 s_1$, a binary number that names the flipped position.

Take the codeword `0110011`. Bit 5 flips, so `0110111` is received. Recompute each check on the received bits $r_1, \ldots, r_7$.

SYNDROME

$$\begin{aligned} s_1 &= r_1 \oplus r_3 \oplus r_5 \oplus r_7 = 0 \oplus 1 \oplus 1 \oplus 1 = 1 \\ s_2 &= r_2 \oplus r_3 \oplus r_6 \oplus r_7 = 1 \oplus 1 \oplus 1 \oplus 1 = 0 \\ s_4 &= r_4 \oplus r_5 \oplus r_6 \oplus r_7 = 0 \oplus 1 \oplus 1 \oplus 1 = 1 \end{aligned}$$

The circles for s_1 and s_4 fail, and s_2 holds. So $s_4 s_2 s_1 = 101$, which is 5 in binary. Flipping bit 5 back gives `0110011`.

$r = 0110111 \qquad s = (s_4 s_2 s_1) = 101 = 5$

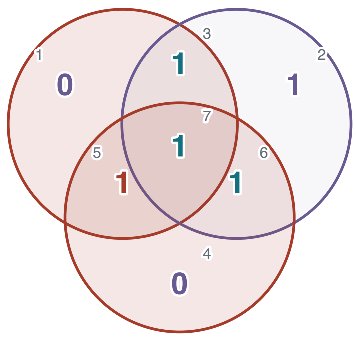


FIGURE 6.36 The received word in the three parity circles, at the syndrome step. The circles of s_1 and s_4 fail (red), and $s_4 s_2 s_1 = 101$ names bit 5.

Sending a source over a channel

A source and a channel can now be joined. Compress the source to $H(U)$ bits a symbol, then code those bits for the channel. Both steps can be made reliable when a channel use carries more than the entropy of the symbol it sends.

SOURCE OVER A CHANNEL

$$H(U) < C \quad \text{bits per channel use}$$

With R_s symbols a second and R_c channel uses a second, compare $H(U)R_s$ with CR_c.

Take a binary source that emits a 1 with probability 0.1, sent over a BSC once a symbol.

A BINARY SOURCE OVER A BSC

$$H_b(0.1) = 0.469 < 1 - H_b(\epsilon) \iff \epsilon < 0.1206 \text{ or } \epsilon > 0.8794$$

At $\epsilon = 0.2$, $C = 1 - H_b(0.2) = 0.278 < 0.469$, so the link cannot be made reliable.

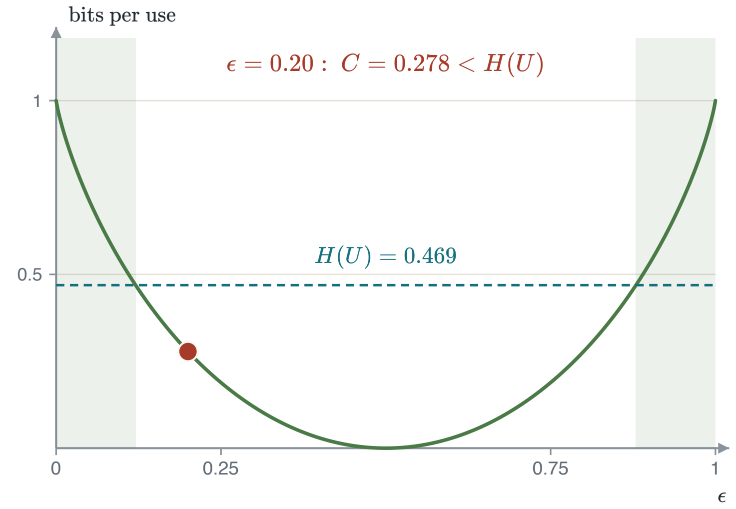


FIGURE 6.37 A binary source with $P(1) = 0.1$ sent over a BSC, one use a symbol, shown at $\epsilon = 0.2$. The green bands mark where $C(\epsilon)$ exceeds $H(U)$.

Capacity around us

Each device below is read as a binary channel with stated numbers. The capacity follows from its matrix.

- A BPSK receiver that decides each bit is a BSC with $p = Q\left(\sqrt{2E/N_0}\right)$, so $C = 1 - H_b(p)$. At 0 dB, $C = 0.603$.
- A network that loses a fraction ϵ of packets, and says which, is an erasure channel: $C = 1 - \epsilon$. At 5% loss, 0.95 of the rate survives.
- A chain of n repeaters, each deciding bits with $p = 0.01$, is one BSC with $p_n = \frac{1}{2}\left(1 - (1-2p)^n\right)$. C falls from 0.919 to 0.559 in 10 hops.
- A memory cell whose charge leaks, so a stored 1 reads as 0 with probability 0.1, is a Z-channel. $C = 0.763$ at $q = 0.54$.

ERASURES ARE CHEAP

A lost packet that is flagged costs ϵ. A flipped bit that is hidden costs $H_b(\epsilon)$, which is more.

HOPS ADD UP

Each decision adds errors, and capacity only falls along a chain.

6.6 The Gaussian channel

The Gaussian channel

The Gaussian channel adds noise to a real-valued input. The input power is limited to P, and the noise Z is Gaussian with power P_N.

GAUSSIAN CHANNEL

$$Y = X + Z, \qquad Z \sim \mathcal{N}(0, P_N), \qquad \mathrm{E}[X^2] \le P$$

Count the codewords that fit, as for the BSC. Over n uses, a received word lies near its codeword, inside a ball of radius $\sqrt{nP_N}$. All received words lie inside a ball of radius $\sqrt{n(P+P_N)}$. Divide the two volumes.

SPHERE COUNT

$$M \approx \frac{\left(\sqrt{n(P+P_N)}\right)^n}{\left(\sqrt{nP_N}\right)^n} = \left(1+\frac{P}{P_N}\right)^{n/2}$$

$$C = \frac{1}{n}\log_2 M = \tfrac{1}{2}\log_2\left(1+\frac{P}{P_N}\right) \quad \text{bits per use}$$

The volume of a ball in n dimensions grows as its radius to the power n, and the constant cancels. At $P/P_N = 15$, $C = \frac{1}{2}\log_2 16 = 2$ bits a use.

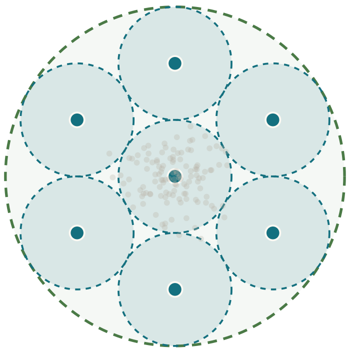


FIGURE 6.38 A received word lands near its codeword, inside a noise disc. All received words lie in a larger disc. The count of noise discs that fit inside is M.

The capacity of the bandlimited channel

A channel of band W carries $2W$ independent samples a second, as in Chapter 1. Each sample is one use of the Gaussian channel. The noise has two-sided density $N_0/2$, so its power in the band is $2W \cdot N_0/2 = N_0 W$.

CAPACITY OF THE BANDLIMITED CHANNEL

$$\begin{aligned} C &= 2W \cdot \tfrac{1}{2}\log_2\left(1+\frac{P}{N_0 W}\right) \\ &= W\log_2(1+\mathrm{SNR}) \quad \mathrm{b/s} \end{aligned}$$

Here $\mathrm{SNR} = P/N_0W$ is a ratio, not a value in decibels: 30 dB is 1000. A band of $W = 1$ MHz at $\mathrm{SNR} = 15$ gives $C = 10^6 \log_2 16 = 4$ Mb/s.

At high SNR, $\log_2(1+\mathrm{SNR}) \approx \log_2 \mathrm{SNR}$. So each doubling of the SNR, or 3 dB, adds one bit a second per hertz.

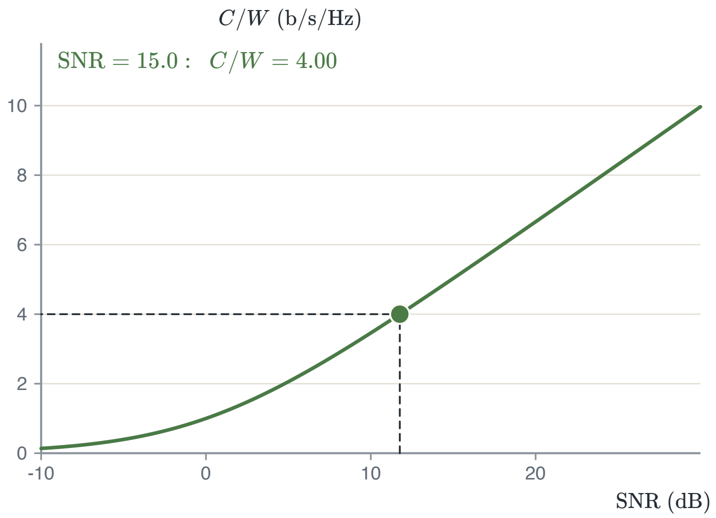


FIGURE 6.39 Capacity per hertz against the SNR in dB, shown at $\mathrm{SNR} = 15$, where $C/W = 4$ b/s/Hz. At high SNR each 3 dB adds one bit a second per hertz.

EXAMPLE 6.4 — A TELEPHONE LINE

GIVEN A telephone line passes 300 Hz to 3.4 kHz at an SNR of 30 dB.

FIND The capacity C.

METHOD The band is $W = 3400 - 300 = 3100$ Hz. The formula $C = W \log_2(1 + \mathrm{SNR})$ takes the SNR as a ratio, so convert it first: $\mathrm{SNR} = 10^{30/10} = 1000$.

SOLUTION $C = 3100 \log_2(1001) = 3100 \times 9.967 = 30.9$ kb/s.

CHECK At 30 dB, $\log_2 1001$ is close to 10, so each hertz carries about 10 bits a second. Then 3.1 kHz carries about 31 kb/s.

COMMON ERROR

Use $\mathrm{SNR} = 1000$, not 30, inside the logarithm. With 30 the answer comes out as $3100 \log_2(1 + 30) = 15.4$ kb/s.

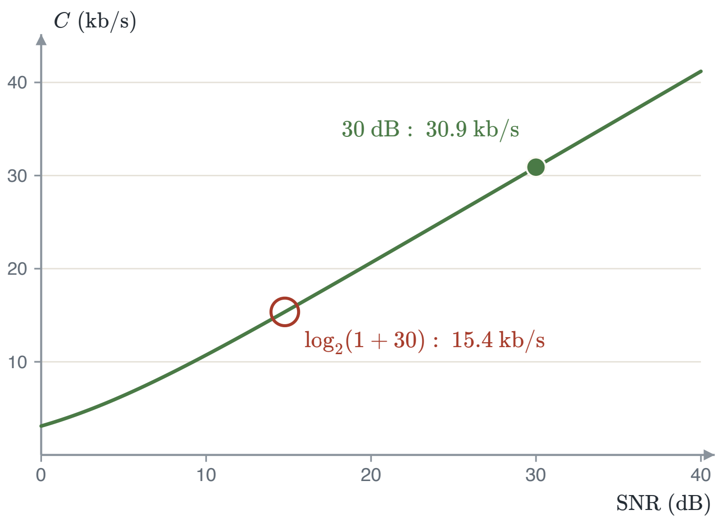


FIGURE 6.40 Capacity of a 3.1 kHz line against its SNR in dB. The dot is the right answer at 30 dB, and the ring is the answer with 30 inside the logarithm.

Capacity against bandwidth

A wider band lets more samples through, but it spreads the same power thinner. The SNR P/N_0W falls as W grows, and the two effects nearly cancel.

For small x, $\ln(1+x) \approx x$, so $\log_2(1+x) \approx x/\ln 2$. Put $x = P/N_0W$ and let W grow.

INFINITE BAND

$$\begin{aligned}\lim_{W\to\infty} W \log_2\left(1 + \frac{P}{N_0W}\right) &= \lim_{W\to\infty} W \cdot \frac{P}{N_0W\ln 2} \\ &= \frac{P}{N_0 \ln 2} = 1.4427\,\frac{P}{N_0}\end{aligned}$$

At a fixed power, capacity levels off however wide the band. At a fixed band it keeps growing with power, but only as the logarithm of the power.

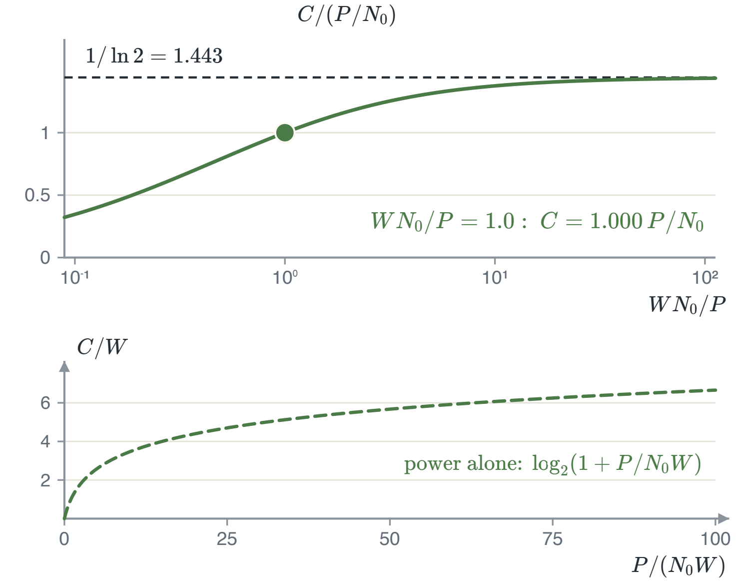


FIGURE 6.41 Top: capacity against bandwidth at a fixed power, levelling off at the dashed line $1.443\,P/N_0$, shown at $WN_0/P = 1$. Bottom: capacity per hertz against power at a fixed band, which keeps rising.

The bandwidth-efficiency plane

The **spectral efficiency** $r = R_b/W$ is the bit rate carried by each hertz of band. The power is energy a bit times bits a second, $P = E_bR_b$. Put both into $R_b < C$.

LEAST E_b/N_0 FOR A SPECTRAL EFFICIENCY

$$\begin{aligned}R_b &< W \log_2\left(1 + \frac{E_bR_b}{N_0W}\right) \\ r &< \log_2\left(1 + r\,\frac{E_b}{N_0}\right) \\ \frac{E_b}{N_0} &> \frac{2^r - 1}{r}\end{aligned}$$

Divide the first line by W to get the second. Raise 2 to the power of each side and solve for E_b/N_0 to get the third. At $r = 2$, $E_b/N_0 > (2^2 - 1)/2 = 1.5$, which is $10\log_{10} 1.5 = 1.76$ dB.

The curve splits the plane into two regions. Above $r = 1$ the band is scarce, and a link is **bandwidth-limited**. Below it power is scarce, and a link is **power-limited**. No link works to the left of the curve.

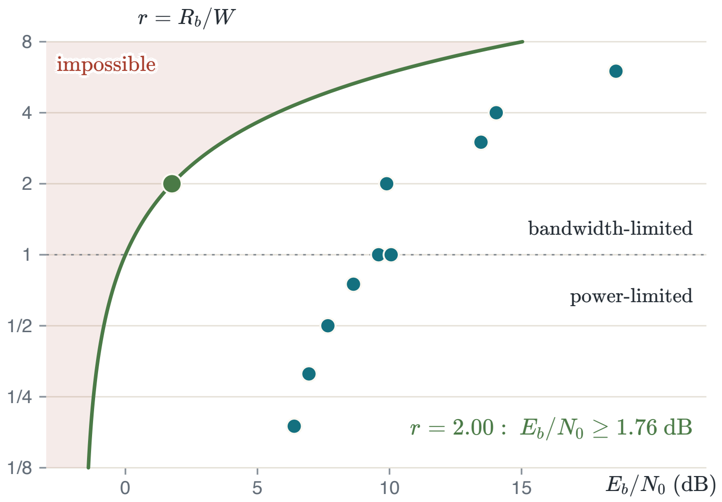


FIGURE 6.42 The least E_b/N_0 for each spectral efficiency r (line), shown at $r = 2$. The cyan dots are the uncoded schemes of Chapter 5 at a symbol error of 10^{-5}.

⚠ THE GAP

Each uncoded scheme of Chapter 5 sits several decibels right of the curve. Channel coding exists to close that gap.

The Shannon limit

Let the spectral efficiency fall to zero, which spends band freely. The bound then falls toward a floor. For small r, $2^r = e^{r \ln 2} \approx 1 + r \ln 2$.

SHANNON LIMIT

$$\frac{E_b}{N_0} > \lim_{r \to 0} \frac{2^r - 1}{r} = \lim_{r \to 0} \frac{r \ln 2}{r}$$
$$= \ln 2 = 0.693 = -1.59 \text{ dB}$$

No code works below -1.59 dB, at any bandwidth. Chapter 5 met the same floor for orthogonal signals as M grows.

Uncoded BPSK needs 9.59 dB for $P_b = 10^{-5}$. Its gap to the limit is $9.59 - (-1.59) = 11.18$ dB. Modern codes come within a fraction of a decibel of the limit.

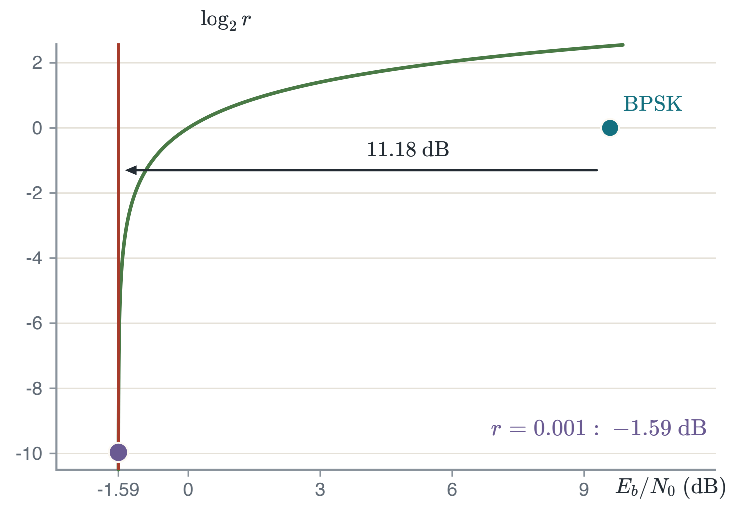


FIGURE 6.43 The bound $(2^r - 1)/r$ as r falls: the point walks down to $\ln 2 = -1.59$ dB. Shown at the last step, with uncoded BPSK at $P_b = 10^{-5}$ placed 11.18 dB to the right.

Water-filling

A total power P must be shared over parallel Gaussian channels with different noise N_i.

PARALLEL CHANNELS

$$C = \sum_i \tfrac{1}{2}\log_2\left(1 + \frac{P_i}{N_i}\right), \qquad \sum_i P_i = P$$

The best share fills every channel up to the same level μ. Channels whose noise lies above it get nothing.

WATER-FILLING

$$P_i = \max(0,\ \mu - N_i)$$

With very little total power, the water first covers the lowest floor. So the quietest channel gets it.

Take six channels with noise $0.1, 0.2, 0.4, 0.8, 1.6, 3.2$ and $P = 1$. Guess that three are used, and solve $\sum_i(\mu - N_i) = P$ for μ.

WATER LEVEL AT $P = 1$

$$3\mu - (0.1 + 0.2 + 0.4) = 1 \implies \mu = \frac{1.7}{3} = 0.567$$

$$P_i = \mu - N_i = 0.467,\ 0.367,\ 0.167$$

The level lies below 0.8, the noise of the fourth channel, so three channels are used.

Each used channel has $1 + P_i/N_i = \mu/N_i$. Add the three capacities.

CAPACITY AT $P = 1$

$$C = \tfrac{1}{2}\log_2 \frac{0.567}{0.1} + \tfrac{1}{2}\log_2 \frac{0.567}{0.2} + \tfrac{1}{2}\log_2 \frac{0.567}{0.4}$$
$$= 1.251 + 0.751 + 0.251 = 2.254 \text{ bits}$$

Equal shares give 1.641 bits.

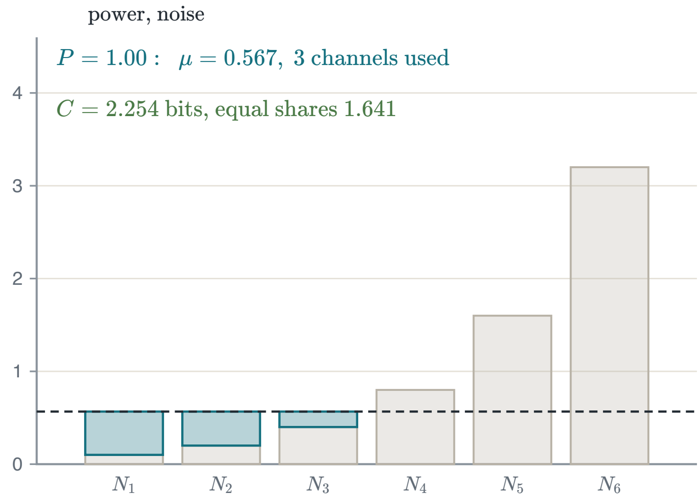


FIGURE 6.44 Six channels with noise 0.1 to 3.2 (grey). Power (cyan) is poured in up to one level μ, shown at $P = 1$.

Shannon's formula around us

$C = W\log_2(1 + P/N_0W)$ takes a width in hertz and a ratio, never a ratio in dB. Four uses of it:

- A telephone line passes 300 Hz to 3.4 kHz. At an SNR of 30 dB, $C = W\log_2(1 + \mathrm{SNR}) = 30.9$ kb/s.
- A radio channel 20 MHz wide at an assumed SNR of 20 dB: $C = 20\log_2 101 = 133$ Mb/s.
- A deep-space link with an assumed $P/N_0 = 10^4$ Hz. More band helps less and less: $C \to P/(N_0 \ln 2) = 14.4$ kb/s.
- DSL splits its band into tones 4.3125 kHz apart. With noise rising in frequency, water-filling gives tone k the bits $b_k = \log_2(1 + P_k/N_k)$.

TWO REGIMES

A wide band at low SNR is power-limited. A narrow band at high SNR is bandwidth-limited.

ASSUMED NUMBERS

The SNR and P/N_0 here are assumptions for the example, not a claim about any standard.

6.7 Summary

From a source to a channel

Follow one source through a Huffman code and a BPSK link, and check that it fits the capacity.

The source $0.4, 0.2, 0.2, 0.1, 0.1$ emits $R_s = 1000$ symbols a second. It has $H = 2.122$ bits a symbol. The Huffman code of Section 6.3 spends $\bar{L} = 2.2$, so 2200 b/s leave the encoder.

The bits go by BPSK at $R_c = 3000$ bits a second and 4 dB, decided bit by bit. That is a BSC with $p = 0.0125$ and $C = 1 - H_b(0.0125) = 0.903$ bit a use.

THE CHECK

$$H\,R_s = 2.122 \times 1000 = 2122 \text{ b/s} \;<\; C\,R_c = 0.903 \times 3000 = 2709 \text{ b/s}$$

A channel code of rate $2200/3000 = 0.733 < C$ exists that makes the link reliable. At $R_c = 2000$ uses a second, $CR_c = 0.903 \times 2000 = 1806 < 2122$ b/s, and the link cannot be made reliable.

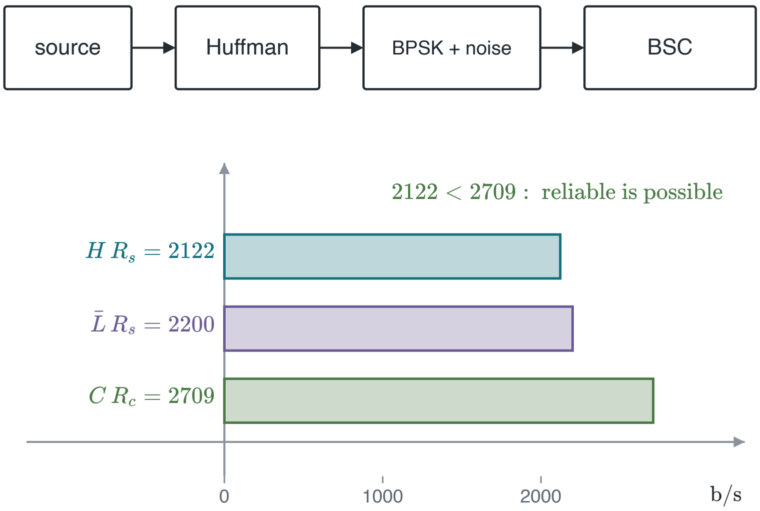


FIGURE 6.45 The source of $0.4, 0.2, 0.2, 0.1, 0.1$ at 1000 symbols a second, a Huffman code, and BPSK at 3000 bits a second and 4 dB. The bars compare the rates.

Quick check

1 Eight equally likely symbols have an entropy of 1 bit, 3 bits or 8 bits?
Answer: 3 bits, since $\log_2 8 = 3$.

2 Codeword lengths $1, 2, 3, 3$ give a Kraft sum of 0.875, 1 or 1.25?
Answer: 1, since $\frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \frac{1}{8} = 1$.

3 A Huffman code for a source of entropy H has $\bar{L}$ below H, in $[H, H+1)$, or above $H+1$?
Answer: In $[H, H+1)$. It meets the source-coding bound.

4 For independent X and Y, is $I(X;Y)$ equal to 0, $H(X)$ or $H(X,Y)$?
Answer: 0, since $H(X \mid Y) = H(X)$.

5 A binary erasure channel with $\epsilon = 0.2$ has capacity 0.2, 0.278 or 0.8?
Answer: 0.8, since $C = 1 - \epsilon$.

6 Below which E_b/N_0 can no code be reliable: -1.59 dB, 0 dB or 9.59 dB?
Answer: -1.59 dB, since $\ln 2 = 0.693$, which is -1.59 dB.

Results of the chapter

TABLE 6.5 Summary of Chapter 6: an introduction to information theory.

QUESTION	RESULT	ANCHOR
How much information does a symbol carry?	$I(s_k) = -\log_2 p_k$ bits: rare symbols carry more.	PS CH12.1.1
What is entropy, and what bounds it?	$H(S) = -\sum_k p_k \log_2 p_k$, with $0 \le H \le \log_2 K$.	PS CH12.1.1
What does an extended source carry?	$H(S^n) = nH(S)$ for a memoryless source.	PS CH12.1.2
What are typical sequences?	About 2^{nH} sequences, each of probability near 2^{-nH}, that hold almost all the probability.	PS CH12.2
When do codeword lengths allow a prefix code?	Exactly when $\sum_k 2^{-l_k} \le 1$.	—
How close can a code get to the entropy?	$H \le \bar{L} < H + 1$, and $H + 1/n$ for blocks of n.	PS CH12.3.1
How is a Huffman code built?	Merge the two least likely entries until one is left, then read each codeword back.	PS CH12.3.1
What does Lempel–Ziv need to know about the source?	Nothing. It parses the stream into new phrases, and its cost falls toward H as the stream grows.	PS CH12.3.2
What is a binary symmetric channel?	Each bit flips with probability p. Hard-decision BPSK gives $p = Q\left(\sqrt{2E_b/N_0}\right)$.	PS CH12.4
What is mutual information?	$I(X;Y) = H(X) - H(X \mid Y)$, symmetric and never negative.	PS CH12.1.3
What is the capacity of a channel?	$C = \max_{p(x)} I(X;Y)$: $1 - H_b(p)$ for the BSC, $1 - \epsilon$ for the erasure channel.	PS CH12.5
What does the channel coding theorem say?	Rates below C can be made reliable. Rates above it cannot.	PS CH12.5
What is the capacity of a bandlimited channel?	$C = W \log_2(1 + P/N_0 W)$ b/s, which levels off at $1.44\, P/N_0$ as W grows.	PS CH12.5.1
What is the Shannon limit?	$E_b/N_0 > \ln 2 = -1.59$ dB. Uncoded BPSK sits 11.2 dB above it.	PS CH12.6

Compress the source to its entropy, then code it for the channel below capacity. The Hamming code of Section 6.5 is a first such channel code.

Projects to try

Four optional projects use the chapter on real text and simulated channels. Each gives an aim, what it practises, a few steps and what to look for.

ⓘ AN ENTROPY METER AND A HUFFMAN CODER

Measure the entropy of a text and code it with a Huffman code built from its own counts.

Practises: Letter frequencies as probabilities. Building a Huffman code by repeated merges. Average length against entropy.

1. Count the letters and spaces of a long text.
2. Compute the entropy of one letter.
3. Build a Huffman code and encode the text.
4. Decode it and check that nothing changed.

Look for: the bits a letter sit just above the entropy, and far above the 1.3 bits of English with its memory.

ⓘ AN LZ78 COMPRESSOR

Compress binary streams with LZ78 and compare the cost with the entropy.

Practises: Parsing a stream into new phrases. A pointer and one new bit a phrase. Why a universal code needs long streams.

1. Draw bits with $P(1) = p$ for several p.
2. Parse each stream and count the phrases.
3. Compute the bits sent a source bit.
4. Plot the cost against the stream length and $H_b(p)$.

Look for: the cost falls slowly toward $H_b(p)$ and never below it.

ⓘ A BSC SIMULATOR

Send bits over a simulated BSC with repetition codes and a Hamming code.

Practises: Majority voting over copies. Syndrome decoding of one error. Rate against error, next to capacity.

1. Flip random bits with probability p.
2. Code with repetition $n = 1, 3, 5$ and with the $(7, 4)$ Hamming code.
3. Decode and count the bit errors.
4. Plot error against rate with $C = 1 - H_b(p)$.

Look for: the Hamming code gives more rate than repetition at a similar error, and all points sit left of C.

WATER-FILLING ON A LINE

Share power over the tones of a line whose noise rises with frequency.

Practises: Parallel Gaussian channels. The water level found by bisection. Why the noisiest tones get nothing.

1. Choose a noise floor that grows with frequency over 64 tones.
2. Find the water level for a total power by bisection.
3. Compute the bits each tone carries.
4. Compare the total with equal shares as the power changes.

Look for: at low power a few quiet tones carry everything. At high power the shares become nearly equal.

APPENDIX

A

Summary of formulas

The results each chapter carries forward, in course order and without derivations.

A.1 The transition from analog to digital — Chapter 1

TABLE A.1 The results of Chapter 1, with $\mathrm{sinc}(x) = \sin(\pi x)/(\pi x)$.

QUESTION	RESULT	SOURCE
What does sampling do to the spectrum?	$G_\delta(f) = f_s \sum_n G(f - nf_s)$: a copy at every multiple of f_s, scaled by f_s.	Ch. 1 · PS CH7.1.1
What is the lowest safe sampling rate?	The Nyquist rate $2W$. Below it the copies overlap and alias.	Ch. 1 · PS CH7.1.1
What filter turns the samples back into the signal?	An ideal lowpass filter with gain T_s. At $f_s = 2W$ its impulse response is $\mathrm{sinc}(2Wt)$.	Ch. 1 · PS CH7.1.1
How is $g(t)$ written in terms of its samples?	$g(t) = \sum_n g(nT_s)\, \mathrm{sinc}\big(2W(t - nT_s)\big)$, exact at every t.	Ch. 1 · PS CH7.1.1
Mid-rise or mid-tread: which has a level at zero?	Mid-tread. Mid-rise has a boundary at zero and outputs $\pm\Delta/2$ there.	Ch. 1 · PS CH7.2.1
What is the power of the quantization noise?	$E[Q^2] = \Delta^2/12$, with $\Delta = 2m_{\max}/L$, for a fine quantizer.	Ch. 1 · PS CH7.2.1
What does one more bit buy?	6.02 dB: $\mathrm{SQNR} = \alpha + 6.02R$, and $\alpha = 1.76$ dB for a full-scale sinusoid.	Ch. 1 · PS CH7.2.1
Why compand?	Small amplitudes are common. Finer steps near zero keep the SQNR steady as the level falls.	Ch. 1 · PS CH7.2.1
What bit rate does PCM need?	$R_b = R\, f_s$. Telephone speech: $8(8000) = 64$ kb/s.	Ch. 1 · PS CH7.3
Why use a Gray code?	Adjacent levels differ in one bit, so a small decision error costs one bit error.	Ch. 1 · PS CH7.3
What bandwidth does PCM need?	$B_T \geq R_b/2$, which is RW at $f_s = 2W$. Each extra bit adds 6.02 dB and W hertz.	Ch. 1 · PS CH7.4.1
What do DPCM and delta modulation send?	The difference from a prediction. Delta modulation sends one bit a sample and needs Δf_s above the largest slope.	Ch. 1 · PS CH7.4.2–7.4.3

A.2 Baseband transmission of digital signals — Chapter 2

TABLE A.2 The results of Chapter 2.

QUESTION	RESULT	SOURCE
Which filter gives the largest peak SNR?	The matched filter $h(t) = k\,g(T-t)$, or $H(f) = k\,G^*(f)\,e^{-j2\pi fT}$, sampled at $t = T$.	Ch. 2 · PS CH8.3.2
What is that largest peak SNR?	$\eta_{\max} = 2E/N_0$. It depends on the pulse energy E, not on its shape.	Ch. 2 · PS CH8.3.2
Where do the two symbols of polar signalling sit?	At $\pm\sqrt{E_b}$ on the axis of the unit-energy $\psi(t)$, a distance $2\sqrt{E_b}$ apart.	Ch. 2 · PS CH8.2.1
When do the correlator and the matched filter agree?	At $t = T_b$, where both give $\int_0^{T_b} x(t)\,\psi(t)\,dt$, for any shape of ψ.	Ch. 2 · PS CH8.3.1
What noise is left in the demodulator output?	$y = s_m + n$, with n Gaussian of mean 0 and variance $N_0/2$.	Ch. 2 · PS CH8.3.1
Where does the optimal threshold go?	$\lambda_{\text{opt}} = \frac{N_0}{4\sqrt{E_b}} \ln \frac{p_0}{p_1}$, at 0 for equal priors and toward the less likely symbol otherwise.	Ch. 2 · PS CH8.3.3
What is $Q(x)$?	The Gaussian tail $Q(x) = \frac{1}{2}\,\mathrm{erfc}(x/\sqrt{2})$, the chance that a unit Gaussian exceeds x.	Ch. 2 · PS CH8.3.3
What is P_b for polar signalling?	$P_b = Q\left(\sqrt{2E_b/N_0}\right)$ with equal priors and a matched filter.	Ch. 2 · PS CH8.3.3
How much interference does an RC channel add?	A bit m bits back adds $(1-q)q^m$, with $q = e^{-2\pi BT_b}$. The worst case totals q.	Ch. 2 · PS CH10.1.1
When does the eye close?	The opening is $2(1-2q)$. It closes at $BT_b = \ln 2/(2\pi) = 0.110$, and errors then occur without noise.	Ch. 2
What is Nyquist's criterion?	$p(kT_b) = \delta[k]$ exactly when $\sum_n P(f - nR_b) = T_b$. It needs $B_T \geq R_b/2$.	Ch. 2 · PS CH10.3.1
What bandwidth does a raised cosine need?	$B_T = \frac{R_b}{2}(1+\alpha)$. The tails fall as $1/\lvert t\rvert^3$ for $\alpha > 0$.	Ch. 2 · PS CH10.3.1

A.3 Geometric representation of signal waveforms — Chapter 3

TABLE A.3 The results of Chapter 3.

QUESTION	RESULT	SOURCE
What is an orthonormal set?	Functions with $\int \psi_j \psi_k \, dt = 1$ for $j = k$ and 0 for $j \neq k$: unit energy, and every pair orthogonal.	Ch. 3 · PS CH8.1
How is a coordinate computed?	$s_{ij} = \int_0^T s_i(t)\, \psi_j(t)\, dt$, one correlator for each basis function.	Ch. 3 · PS CH8.1
How is the waveform rebuilt?	$s_i(t) = \sum_{j=1}^{N} s_{ij}\, \psi_j(t)$. The vector $\mathbf{s}_i$ holds the whole waveform.	Ch. 3 · PS CH8.1
What does the translation preserve?	Inner products: $\int x(t)\, y(t)\, dt = \sum_k x_k y_k$.	Ch. 3 · PS CH8.1
What is the energy of a signal?	$E_i = \lVert \mathbf{s}_i \rVert^2$, the squared distance from its point to the origin.	Ch. 3 · PS CH8.1
What is the distance between two signals?	$d_{ik} = \lVert \mathbf{s}_i - \mathbf{s}_k \rVert$, the root of $\int (s_i - s_k)^2 \, dt$.	Ch. 3 · PS CH8.1
Antipodal or orthogonal: which is further apart?	Antipodal, $d = 2\sqrt{E_b}$ against $\sqrt{2E_b}$. Orthogonal needs twice the energy, 3 dB, for the same distance.	Ch. 3 · PS CH8.2
When are a cosine and a sine orthogonal?	Over $[0, T]$ when $f_c T$ is a whole number. Scaled by $\sqrt{2/T}$ they are an orthonormal pair.	Ch. 3 · PS CH8.6.1
How far apart are the points of M-PSK?	$d_{\min} = 2\sqrt{E} \sin(\pi/M)$. All M points lie on the circle of radius $\sqrt{E}$.	Ch. 3 · PS CH8.6.1
What is one Gram–Schmidt step?	$g_k = s_k - \sum_{i<k} s_{ki} \psi_i$, then $\psi_k = g_k / \sqrt{E_{g_k}}$. A zero g_k adds no axis.	Ch. 3 · PS CH8.1
How many axes does a set of M signals need?	$N \leq M$, with $N = M$ only when no signal is a combination of the others.	Ch. 3 · PS CH8.1
What does another basis change?	Only the coordinates. N, the energies and the distances stay, and so do the receiver and its error probability.	Ch. 3 · PS CH8.1

A.4 The optimal receiver in additive white Gaussian noise — Chapter 4

TABLE A.4 The results of Chapter 4.

QUESTION	RESULT	SOURCE
How many bits does one of M waveforms carry?	$k = \log_2 M$, so $R_b = kR_s$ and $T_b = T/k$.	Ch. 4 · PS CH8.4
What does the correlator bank give?	$\mathbf{r} = \mathbf{s}_i + \mathbf{n}$, with $r_j = \int_0^T r(t)\,\psi_j(t)\,dt$.	Ch. 4 · PS CH8.4.1
When does a matched filter give the same number?	At the sampling time $t = T$. Its impulse response is $\psi_j(T-t)$.	Ch. 4 · PS CH8.4.1
Why can the noise outside the signal space be dropped?	It does not depend on the signal and adds the same $\|n'\|^2$ to every distance.	Ch. 4 · PS CH8.4.1
What are the noise components?	Independent Gaussians with mean 0 and variance $N_0/2$. The cloud is round.	Ch. 4 · PS CH8.4.1
What does MAP maximise?	$P(\mathbf{s}_i)\,f(\mathbf{r} \mid \mathbf{s}_i)$. With equal priors it is ML.	Ch. 4 · PS CH8.4.1
What is ML in Gaussian noise?	Minimum distance. MAP subtracts $N_0 \ln P(\mathbf{s}_i)$ from each squared distance.	Ch. 4 · PS CH8.4.1
What is the correlation metric?	$\mathbf{r} \cdot \mathbf{s}_i - E_i/2 + \frac{N_0}{2} \ln P(\mathbf{s}_i)$. Keep the largest.	Ch. 4 · PS CH8.4.1
What shape are the decision regions?	Convex polygons cut by bisectors. Unequal priors shift each line toward the less likely point.	Ch. 4 · PS CH8.4.1
What is the binary error probability?	$Q\left(\sqrt{d^2/2N_0}\right)$ for equal priors.	Ch. 4 · PS CH8.3.3, 8.4.2
What is the union bound?	$P_e \leq \frac{1}{M} \sum_i \sum_{j \neq i} Q\left(\sqrt{d_{ij}^2/2N_0}\right)$. The intelligent bound keeps only faces.	Ch. 4 · PS CH8.4.2
What is the nearest-neighbour form?	$\bar{N}_{\min} Q\left(\sqrt{d_{\min}^2/2N_0}\right)$. Close at high SNR, but not a bound.	Ch. 4 · PS CH8.4.2

A.5 Digital modulation methods — Chapter 5

TABLE A.5 The results of Chapter 5.

QUESTION	RESULT	SOURCE
What does a carrier do to the spectrum?	$U(f) = \frac{1}{2}S(f - f_c) + \frac{1}{2}S(f + f_c)$. The band doubles to $2W$ and the energy halves.	Ch. 5 · PS CH8.5.1
What does the IQ modulator send?	$s(t) = I\cos(2\pi f_c t) - Q\sin(2\pi f_c t)$: one point of the plane a symbol.	Ch. 5 · PS CH8.6, 8.7
How do the binary schemes compare?	BPSK: $Q\left(\sqrt{2E_b/N_0}\right)$. BFSK and BASK: $Q\left(\sqrt{E_b/N_0}\right)$, 3 dB worse.	Ch. 5 · PS CH8.3.3, 8.6.1, 9.5
What is the M-PSK neighbour distance?	$d_{\min} = 2\sqrt{E_s}\sin(\pi/M)$, and $P_e \approx 2Q\left(\sqrt{2E_s/N_0}\sin(\pi/M)\right)$.	Ch. 5 · PS CH8.6.1, 8.6.3
What do Gray labels buy?	Neighbours differ in one bit, so $P_b \approx P_e/\log_2 M$.	Ch. 5 · PS CH8.6.1, 8.6.3
What does DPSK save, and what does it cost?	No carrier phase. $P_b = \frac{1}{2}e^{-E_b/N_0}$, under 1 dB behind BPSK.	Ch. 5 · PS CH8.6.4, 8.6.5
What is the M-ASK error?	$\frac{2(M-1)}{M}Q\left(\sqrt{6E_s/((M^2-1)N_0)}\right)$, about 6 dB a doubling.	Ch. 5 · PS CH8.5.3
What is the square-QAM error?	$4(1 - 1/\sqrt{M})\,Q\left(\sqrt{3E_s/((M-1)N_0)}\right)$, with $E_s = (M-1)d^2/6$.	Ch. 5 · PS CH8.7.3
How much does QAM save over PSK?	4.20 dB at $M = 16$ and 9.95 dB at $M = 64$.	Ch. 5 · PS CH8.7.3
What do M orthogonal signals give?	$M - 1$ neighbours at $\sqrt{2E_s}$. The E_b/N_0 needed falls as M grows, toward -1.6 dB.	Ch. 5 · PS CH9.1.1, 9.1.2
What does noncoherent BFSK need?	Tones $1/T$ apart and an envelope detector. $P_b = \frac{1}{2}e^{-E_b/2N_0}$.	Ch. 5 · PS CH9.5.2, 9.5.3
What is MSK?	BFSK at $\Delta f = 1/(2T_b)$ with a continuous phase. Its envelope is constant.	Ch. 5 · PS CH9.6.1, 9.6.2
How wide is each family?	PSK and QAM: $R_b/\log_2 M$. Orthogonal: $MR_b/(2\log_2 M)$.	Ch. 5 · PS CH10.2, 9.7
Where does each family sit on the plane?	PAM, PSK and QAM: bandwidth-limited, $r > 1$. Orthogonal: power-limited, $r < 1$.	Ch. 5 · PS CH9.7

A.6 An introduction to information theory — Chapter 6

TABLE A.6 The results of Chapter 6.

QUESTION	RESULT	SOURCE
How much information does a symbol carry?	$I(s_k) = -\log_2 p_k$ bits: rare symbols carry more.	Ch. 6 · PS CH12.1.1
What is entropy, and what bounds it?	$H(S) = -\sum_k p_k \log_2 p_k$, with $0 \le H \le \log_2 K$.	Ch. 6 · PS CH12.1.1
What does an extended source carry?	$H(S^n) = nH(S)$ for a memoryless source.	Ch. 6 · PS CH12.1.2
What are typical sequences?	About 2^{nH} sequences, each of probability near 2^{-nH}, that hold almost all the probability.	Ch. 6 · PS CH12.2
When do codeword lengths allow a prefix code?	Exactly when $\sum_k 2^{-l_k} \le 1$.	Ch. 6
How close can a code get to the entropy?	$H \le \bar{L} < H + 1$, and $H + 1/n$ for blocks of n.	Ch. 6 · PS CH12.3.1
How is a Huffman code built?	Merge the two least likely entries until one is left, then read each codeword back.	Ch. 6 · PS CH12.3.1
What does Lempel–Ziv need to know about the source?	Nothing. It parses the stream into new phrases, and its cost falls toward H as the stream grows.	Ch. 6 · PS CH12.3.2
What is a binary symmetric channel?	Each bit flips with probability p. Hard-decision BPSK gives $p = Q\left(\sqrt{2E_b/N_0}\right)$.	Ch. 6 · PS CH12.4
What is mutual information?	$I(X;Y) = H(X) - H(X \mid Y)$, symmetric and never negative.	Ch. 6 · PS CH12.1.3
What is the capacity of a channel?	$C = \max_{p(x)} I(X;Y)$: $1 - H_b(p)$ for the BSC, $1 - \epsilon$ for the erasure channel.	Ch. 6 · PS CH12.5
What does the channel coding theorem say?	Rates below C can be made reliable. Rates above it cannot.	Ch. 6 · PS CH12.5
What is the capacity of a bandlimited channel?	$C = W \log_2(1 + P/N_0 W)$ b/s, which levels off at $1.44\, P/N_0$ as W grows.	Ch. 6 · PS CH12.5.1
What is the Shannon limit?	$E_b/N_0 > \ln 2 = -1.59$ dB. Uncoded BPSK sits 11.2 dB above it.	Ch. 6 · PS CH12.6

Digital Communications · Lecture Notes

VERSION	DATE	CHANGES
v0	24 September 2026	First edition.