Some Digital Communication Fundamentals for Physicists and Others

David G. Messerschmitt

Electrical Engineering and Computer Sciences
University of California at Berkeley

Technical Report No. UCB/EECS-2008-78
http://www.eecs.berkeley.edu/Pubs/TechRpts/2008/EECS-2008-78.html

June 6, 2008

Some Digital Communication Fundamentals for Physicists and Others

David G. Messerschmitt
messer@eecs*
Department of Electrical Engineering and Computer Sciences
University of California at Berkeley

June 4, 2008

*Add berkeley.edu

Contents

1 Introduction

The purpose of this report is to concisely describe some fundamentals of digital communication in a simple and accessible way. An emphasis is placed on intuition with an audience of non-specialists in mind. Electronic communication as a separate discipline has a nine-decade history, and even a concise summary of the current state of knowledge would take thousands of pages. Nevertheless, underlying all this work are important and lasting principles that form a solid foundation for communication system design. Since these principles are very fundamental, most were actually appreciated by the 1950's.

As an example of the intended readership, in reviewing some of the physics literature, we find limited awareness of some fundamental principles of communications. Such awareness would be valuable in physics research that may have electronic or photonic communication application, such as new modes of electromagnetic propagation and new propagation media.

We do not cover optical communication, a vast field in its own right and one where advances are more dominated by underlying physical principles than they are by communication principles. There are many books on aspects of communications, but a couple of recommended general texts are [1, 2]. Analog communications (such as AM and FM radio) are rare in new communication system design and are also not covered.

The notation of probability and random processes is reviewed in Appendix A. Please refer to this Appendix if notational questions arise.

2 Channels

Communication theory often utilizes a model for a communication channel as a transformation from a real-valued input signal $x(t)$ to a real-valued output signal $y(t)$. Such a model allows a separation of concerns. The characteristics of the physical medium, antennas, modulators, etc., can be embedded within the channel model, and the communication principles can take the simple input-output model as a starting point and attempt to extract the greatest performance possible within the domain of the channel model. These channel models are usually very accurate, and as a result most advances in non-optical communication systems of the past five decades have proceeded on the basis of theory and simulation rather than experiment.

2.1 Noisy channels

Due to the prevalence of noise as a factor that limits communication performance, the channel model is usually statistical, in which case the input and output are random processes $X(t)$ and $Y(t)$. For example, a simple model that accurately fits a number of situations is

$$Y(t) = X(t) \otimes g(t) + N(t) \tag{1}$$

and where $\otimes$ denotes the convolution operator, defined as

$$r(t) \otimes s(t) = \int r(\tau) \cdot s(\tau - t)\, \mathrm{d}\tau\,. \tag{2}$$

This convolution is one mathematical description of a linear time-invariant filter that has *impulse response* $g(t)$.

Although virtually all physical channels are continuous-time as in (1), it is common to embed sampling within the channel model[1]. This results in a discrete time model, as in

$$Y_k = X_k \otimes g_k + N_k\,, \tag{3}$$

where discrete-time convolution operator is defined as

$$r_k \otimes s_k = \sum_m r_m \cdot s_{k-m} \tag{4}$$

Of course these simple models are often inadequate, for example when the channel includes time-varying effects (e.g. due to relative motion of transmitter and receiver) or the channel is multiple-input multiple-output (MIMO) (as when there are mulitiple antennas in receiver or transmitter). The MIMO case is considered in Section 4.3.

2.2 Noise

Even when the channel is not frequency selective ($g(t) = \delta(t)$ or $g_k = \delta_k$), there are at least two obstacles to reliable communication: noise and signal power. If there is noise but the signal power is unbounded, reliable communication becomes easy. Or if the signal power is fixed but the noise goes to zero, reliable communication also becomes easy. In the continuous-time case, we can even communicate reliably at arbitrary high speeds under either of these limiting conditions. In this sense, signal power and channel noise thus limit on our ability to communicate. In exactly what sense they become a limitation will be considered in detail later, but let us summarize the conclusion. Power limits and noise do *not* impair our ability to communicate reliably, but they do place a

[1]The sampling theorem suggestions that there is often nothing lost in doing so, as long as the continuous-time channel is bandlimited.

limit on the *rate* at which we can communicate. For most of us, this conclusion would be counter-intuitive. In the absence of any bandlimit though the channel, intuition would tend to suggest that we could communicate at arbitrary high rates, but that noise would prevent completely reliable communication.

Assume the signal power is limited by practical considerations like the power of the transmit electronics or the energy bill. Then in most applications, it is imperative to operate at the limits of what the noise allows, which we might call the "noise floor", at least under the worst case conditions for a given design. Thus, there is great interest in extending performance as far as possible, possibly subject to cost constraints. For example, cable systems require expensive repeaters, motivating us to use as few as possible, as well as limits on the power that can be supplied, motivating us to use relatively low transmit power. Thus, the transmit power is fixed and the spacing between repeaters is stretched until we are operating at the noise floor[2]. Similarly, communication with spacecraft puts a premium on antenna size, weight, and power consumption, motivating us to operate at the noise floor limits.

A common statistical model for noise is a Gaussian random process. This is an accurate model wherever noise is actually a superposition of many micro phenomena, as in thermal noise in a receiver, the background radiation of the universe, or the superposition of multiple interferers. In this case the probability density function of a single sample of the noise is Normal or Gaussian with mean value μ_N and variance σ_N^2 is

$$p_N(n) = \frac{1}{\sigma_N \sqrt{2\pi}} e^{-(n-\mu_N)^2/2\sigma_N^2}\,. \tag{5}$$

Fortuitously a Gaussian noise assumption is usually the most mathematically tractable. In a fundamental sense, Gaussian statistics are also the worst case for a given noise power.

Wireless communication is a different case, in that the most importance source of "noise" is actually interference from other users of the same spectrum and the fact that the signal suffers fading (time variable amplitude and phase) due to multipath effects, resulting in a signal (as well as noise and interfernece) that is random. For example, in terrestrial cellular systems the same frequency bands are reused in different spatial cells, resulting in interference from adjacent cells, and multiple copies of the signal arrive at the receive antenna with slightly difference delays due to bounces off of the ground and buildings and other obstacles. In this case, there is motivation to reduce the transmit power as much as possible, again operating near the noise floor, so as to minimize interference into other systems. In many cases interference can be adequately modeled as Gaussian noise. Also, in the presence of relative transmitter/receiver motion the channel model is fairly rapidly time varying.

[2]The signal level on any cable medium falls as $e^{-\alpha D}$ where D is the distance. Radio propagation in freespace has a much more favorable D^{-2} dependence, and in terrestrial cellular systems due to ground scattering the dependence is $\approx D^{-4}$.

2.3 Passband signals

Analog signals carried by a propagation medium are either baseband (their spectrum includes frequency $f = 0$) or passband. Radio channels are always passband, but often so are cable media, especially where many services are sharing the same medium using frequency separation.

A passband channel can be modeled as an equivalent baseband channel, but one for which the signals are complex-valued. In particular, any real-valued passband signal $u(t)$ can be represented in the form

$$u(t) = \mathrm{Re}\left\{v(t) \cdot e^{i2\pi f_0 t}\right\} \tag{6}$$

where f_0 is a carrier frequency and $v(t)$ is a complex-valued baseband signal (centered at $f = 0$) known as the *complex envelope.*

If $v(t) = a(t) + i \cdot b(t)$ is expressed in terms of real-valued signals $a(t)$ and $b(t)$, known respectively as the *in-phase* and *quadrature* signal components, then (6) can be written as

$$u(t) = a(t) \cdot \cos(2\pi f_0 t) - b(t) \cdot \sin(2\pi f_0 t)\,. \tag{7}$$

When $b(t) \equiv 0$ then this is known as *amplitude modulation*, and when $|v(t)|$ is constant this is known as *phase modulation.* In practice in today's designs neither pure amplitude nor pure phase modulation is very common, but rather the amplitude and phase are manipulated simultaneously.

When $v(t)$ is real-valued, but only then, the spectrum of $u(t)$ is conjugate-symmetric about the carrier frequency. In this case the spectrum is used inefficiently because the lower sideband is the conjugate of the upper sideband, or equivalently only one real-valued signal is conveyed whereas two is feasible.

At the receiver, the complex envelope can be recovered from the passband signal by observing that

$$2\,u(t) \cdot e^{-i2\pi f_0 t} = v(t) + v^*(t) \cdot e^{-i4\pi f_0 t}\,. \tag{8}$$

Thus $v(t)$ can be recovered from $u(t)$ by lowpass filtering $2\,u(t) \cdot e^{-i2\pi f_0 t}$ to remove the $2f_0$ frequency component.

The channel model can be expressed in terms of equivalent complex envelope signals at input and output. This just means that the modulation and demodulation (including lowpass filtering) are embedded within the channel model. In this worldview, the channel input is a complex-valued baseband signal $x(t)$ and the channel output is a complex-valued baseband signal $y(t)$. When noise and channel dispersion are taken into account, the result is something like (3) where $N(t)$ and $g(t)$ are complex-valued as well.

3 Matched filters

Consider the simple case of a received baseband signal of the form

$$Y(t) = A \cdot h(t) + N(t) \tag{9}$$

where $h(t)$ is a waveform with finite energy ε_h^2,

$$\varepsilon_h^2 \doteq \int |h(t)|^2 \,\mathrm{d}t < \infty \,. \tag{10}$$

This is called "known signal waveform in additive noise". The single amplitude A might represent information, such as for example representing m bits of information by choosing A to take on one of 2^m complex values. (A more interesting case is where a whole stream of information is represented by a stream of amplitude values, a generalization to be considered later.)

In this simple case, the goal is to estimate the value A in the receiver. How to do this? One common method is to cross-correlate this reception against the known pulse shape $h(t)$, creating a statistic Z,

$$\begin{aligned} Z &= \int Y(t) \cdot h^*(t) \,\mathrm{d}t \\ &= A \cdot \varepsilon_h^2 + \int N(t) \cdot h^*(t) \,\mathrm{d}t \end{aligned} \tag{11}$$

Note that Z summarizes the entire reception $\{Y(t),\ -\infty < t < \infty\}$ in a single complex-valued number.

There are several ways to justify using Z as the representation of the reception appropriate as the first step in estimating A. First, we can ask what $\hat{A}$ gives the best match to $Y(t)$ in the sense that it minimizes the energy between a candidate signal $\hat{A} \cdot h(t)$ and $Y(t)$,

$$
\begin{aligned}
\epsilon^2 &= \int \left| Y(t) - \hat{A} \cdot h(t) \right|^2 \mathrm{d}t \\
&= \int |Y(t)|^2 \, \mathrm{d}t - \hat{A}^* Z - Z^* \hat{A} + \varepsilon_h^2 \cdot |\hat{A}|^2 \\
&= \int |Y(t)|^2 \, \mathrm{d}t - \frac{|Z|^2}{\varepsilon_h^2} + \frac{\left| \varepsilon_h^2 \cdot \hat{A} - Z \right|^2}{\varepsilon_h^2} .
\end{aligned}
\tag{12}
$$

We would like to minimize ϵ^2 by the choice of $\hat{A}$. Only the last term is a function of $\hat{A}$, and that term is minimized by choosing $\hat{A} = Z/\varepsilon_h^2$. Thus, the estimate $\hat{A} = Z/\varepsilon_h^2$ yields the smallest error energy among all linear estimator structures.

A second justification arises for the simple case where $N(t)$ is white Gaussian noise. The single complex-valued random variable Z is a *sufficient statistic* for the estimation of A, meaning roughly that no information relevant to A is lost in reducing $\{Y(t), -\infty < t < \infty\}$ to a single complex-valued statistic Z. The intuitive reason for this is that the information about $Y(t)$ that is discarded by the matched filter (a) contains no direct information about A (is noise-related only) and (b) is uncorrelated with the noise component of Z and thus statistically independent of that noise component[3]. It is thus irrelevant, since it tells us nothing useful about either the signal or noise components of Z.

An equivalent representation for (11) is

$$
Z = Y(t) \otimes h^*(-t)|_{t=0} \tag{13}
$$

where $\otimes$ is again the convolution operator of (2). This representation in terms of a filter with impulse response $h^*(-t)$, known as a *matched filter*, followed by taking a sample a $t = 0$. In the frequency domain, this is equivalent to multiplying by transfer function $H^*(f)$, where $H(f)$ is the Fourier transform of $h(t)$.

A third justification for the matched filter is intuitive, and best seen in the frequency domain and viewing the effect of the matched filter $H^*(f)$ on phase and magnitude spectra separately:

- The resulting signal response $H(f) \cdot H^*(f) = |H(f)|^2$ at the matched filter output has zero phase. Thus, the matched filter is a perfect phase equalizer.
- The noise power spectrum at the output of the matched filter is $N_0 \cdot |H^*(f)|^2 = N_0 \cdot |H(f)|^2$. Note that if we were to change the phase response of the matched filter, this would have no effect on the noise power spectrum, and hence the noise power. Perfect phase equalization comes at no expense in terms of noise power.

[3] Recall that Gaussian random variables that are uncorrelated are also independent.

- In terms of the magnitude spectrum, with energy ε_h^2 to devote to the filter response $|H(f)|$, how should it be allocated by frequency f? It makes sense to more heavily weight the frequencies where signal $H(f)$ is larger (the local signal-to-noise ratio is larger) than frequencies where $H(f)$ is smaller. This the matched filter does.

3.1 Signal-to-noise ratio

Signal-to-noise ratio, defined as signal power divided by noise power, is often used as a characterization of communication reliability. One justification for this is that in Gaussian noise the first- and second-order (power-like) statistics are a full characterization of the distribution[4]. Caution needs to be exercised, however, since the appropriate definition of SNR depends on the circumstance if SNR is to be meaningful as a performance metric or predictor. We will illustrate this by comparing two circumstances, coherent and incoherent detection.

Example (coherent case). If the signal waveform $h(t)$ is known, then a correlator against the known signal (or equivalently matched filter and sampler) can be used. In this case, the SNR should be calculated *after* the correlator, because this is where post-processing will be applied and the correlator affects the noise as well as signal.

While we are at it, let's show that among all possible cross-correlators, cross-correlation with the known signal waveform $h(t)$ maximizes the SNR. This confirms by a different route that the matched filter is the best processing to use in the receiver in the presence of white Gaussian noise. Replace $h(t)$ by a (possibly) different waveform $g(t)$ in (11),

$$\begin{aligned} Z &= \int Y(t) \cdot g^*(t)\,\mathrm{d}t \\ &= A \int h(t) \cdot g^*(t)\,\mathrm{d}t + \int N(t) \cdot g^*(t)\,\mathrm{d}t\,. \end{aligned} \tag{14}$$

The second term is a Gaussian random variable, and it is readily calculated that if $N(t)$ is white Gaussian noise with power spectral density N_0, then the noise term has mean zero and variance $N_0 \cdot \varepsilon_g^2$. The SNR is defined as the magnitude-squared of the first (signal) term divided by the variance (power) of the second (noise) term,

[4]Of course, if the noise is non-Gaussian, or the signal is somehow stochastic or time-varying, we must dig deeper. In these cases, SNR is inadequate as a characterization of reliability.

$$\begin{aligned} \text{SNR}_{\text{coherent}} &= \frac{\left|A \int h(t) \cdot g^*(t)\,\mathrm{d}t\right|^2}{N_0 \cdot \varepsilon_g^2} \\ &= \frac{|A|^2}{N_0 \cdot \varepsilon_g^2} \cdot \left| \int h(t) \cdot g^*(t)\,\mathrm{d}t \right|^2 . \end{aligned} \tag{15}$$

The subscript ”coherent” refers to the fact that we are assuming full knowledge of the waveform $h(t)$ and taking full advantage of that knowledge.

By the Schwarz inequality,

$$\left| \int h(t) \cdot g^*(t)\,\mathrm{d}t \right|^2 \leq \varepsilon_h^2 \cdot \varepsilon_g^2 \tag{16}$$

with equality if and only if $g(t) = B \cdot h(t)$ for any complex constant B. This implies that

$$\text{SNR}_{\text{coherent}} \leq |A|^2 \cdot \frac{\varepsilon_h^2}{N_0} \,, \tag{17}$$

with equality if and only if the filter is a matched filter ($g(t) = h(t)$). As promised, the matched filter maximizes $\text{SNR}_{\text{coherent}}$, and any other filter results in a lower $\text{SNR}_{\text{coherent}}$. $\text{SNR}_{\text{coherent}}$ for a matched filter numerically equals the energy in the received waveform (proportional to joules) divided by the noise spectral density (proportional to watts/Hz, which is also joules).

Sometimes the signal power is more important than the signal energy. This might occur where interference into other systems is important, or where the power available to the transmitter is limited. Suppose $h(t)$ has time duration T; then the signal power is $P_h = \varepsilon_h^2/T$ and the SNR becomes

$$\text{SNR}_{\text{coherent}} \leq |A|^2 \cdot \frac{P_h \cdot T}{N_0} \,, \tag{18}$$

With fixed signal power, $\text{SNR}_{\text{coherent}}$ increases in proportion to T because there is greater “integration time” to accumulate signal energy in the receiver.

Example (incoherent case). Suppose that we have no knowledge of the waveform $h(t)$, and the detection or estimation technique is to simply estimate the total power in $Y(t)$ and compare it to the expected noise power. We call this incoherent detection, since it relies only on power estimates[5].

If $h(t)$ has simultaneously[6] time duration T and bandwidth W, then the received signal power is $|A|^2 \cdot \varepsilon_h^2/T$ and the noise power within bandwidth W is $N_o \cdot W$, leading to an alternative definition of SNR as

$$\mathrm{SNR}_{\mathrm{incoherent}} = \frac{|A|^2 \cdot \varepsilon_h^2/T}{N_0 \cdot W} = \frac{1}{WT} \cdot \mathrm{SNR}_{\mathrm{coherent}} \,. \tag{19}$$

A comparison of $\mathrm{SNR}_{\mathrm{coherent}}$ from (17) and $\mathrm{SNR}_{\mathrm{incoherent}}$ from(19) and the assumptions behind them gives considerable insight:

1. $\mathrm{SNR}_{\mathrm{coherent}}$ is larger than $\mathrm{SNR}_{\mathrm{incoherent}}$ by a factor WT. This increase in SNR can be attributed to knowledge of waveform $h(t)$ and an estimation or detection technique (the matched filter) that takes full advantage of that knowledge.
2. The matched filter in coherent detection gathers up or integrates the entire signal *energy*, whereas the incoherent detector is fundamentally a *power* estimator.
3. With incoherent detection, increased signal bandwidth is deleterious to detection because total noise power increases with bandwidth. With coherent detection, increasing signal bandwidth has no impact one way or another because the matched filter successfully rejects the additional noise.
4. The significance of signal duration T depends entirely on whether transmit energy or power is considered to be a scarce resource. If signal power is held constant, increasing T assists coherent detection (because at constant power the energy increases in proportion to T) but increasing T but has no effect on incoherent detection (because the SNR is proportional to power rather than energy). If signal energy is held constant, increasing T has no effect on the coherent detection but harms incoherent detection (because SNR is proportional to signal power, which decreases).
5. All these conclusions apply to additive white Gaussian noise. These issues must be revisited when the noise has other statistics.

[5] This definition is common in the physics literature. For example, in the search for extraterrestrial intelligence (SETI) [3] it may be unreasonable to assume knowledge of the waveform $h(t)$ due to lack of coordination of transmitter and receiver.

[6] Technically this is impossible, since a bandlimited signal cannot be time limited and vice versa. However, this can be approximately achieved with increasing accuracy as $WT \to \infty$.

3.2 Pulse-amplitude modulation

As a generalization of (9), a common modulation technique for digital communication is to assume a baseband signal of the form

$$Y(t) = \sum_k A_k \cdot h(t - kT) + N(t)\,. \tag{20}$$

This modulation scheme is called *pulse-amplitude modulation* (PAM). Here is some common terminology associated with PAM:

Symbol interval T in seconds.

Symbol rate $1/T$ in symbols per second.

Data symbols A_k, $-\infty < k < \infty$ are complex values conveying information.

Symbol alphabet is the set from which A_k is drawn. Let the alphabet be $\{a_k,\ 1 \le k \le K\}$. For example, if m bits of information are conveyed with each symbol, then the cardinality of the alphabet would minimally be $K = 2^m$.

Signal constellation is the arrangement of the symbol alphabet on the complex plane. Figure 1 shows a possible signal constellation for an alphabet of cardinality $K = 64$, conveying $m = 6$ bits of information. In this constellation, the alphabet is laid out in a square array.

Bit rate is m/T bits per second, which is what the "customer" cares about[7].

The goal in the choice of the $h(t)$ is to make it orthogonal to its symbol-interval translates,

$$\int h(t) \cdot h^*(t - kT)\,\mathrm{d}t = \varepsilon_h^2 \cdot \delta_k\,. \tag{21}$$

When this orthogonality condition is satisfied, it allows us to build a receiver consisting of a matched filter and symbol rate sampler, which yields a signal component

$$Y(t) \otimes h^*(-t)|_{t=mT} = A_m \cdot \varepsilon_h^2 + N_m\,. \tag{22}$$

The resulting data symbol can be detected by applying thresholds appropriate for the signal constellation to this noisy estimate of the data symbol. Often (21) is satisfied at the transmitter, but violated after passing through a dispersive channel. The resulting

[7]It has become commonplace to call this "bandwidth" in networking, as in "my DSL line gives me a bandwidth of 1.5 megabits per second". This reflects the ignorance of communication on the part of networking specialists, because bandwidth W is something entirely different.

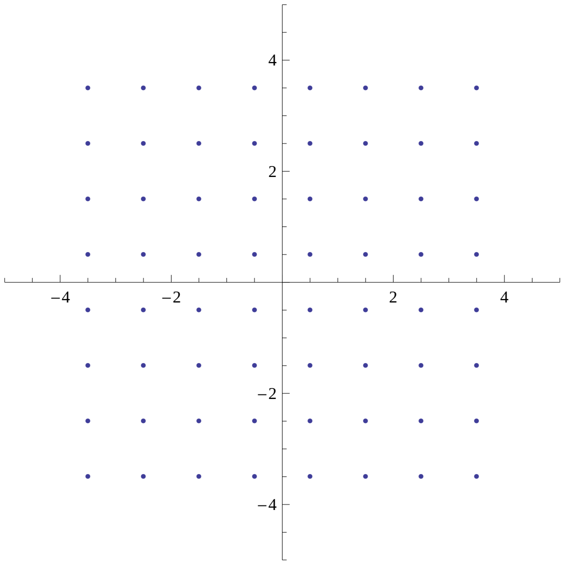

Figure 1: This 8×8 signal constellation has an alphabet with cardinality 64, and hence can be used to communicate six bits of information per symbol.

impairment is known as *intersymbol interference (ISI)*, a topic that has been the subject of extensive research[8].

An important property of a signal constellation is the average energy,

$$\begin{aligned}\varepsilon_a^2 &\doteq E_A|A_k|^2 \\ &= \sum_{k=1}^{K} |a_k|^2 \cdot p_A(a_k)\end{aligned} \tag{23}$$

assuming, of course, knowledge of the probability mass function $p_A(a_k)$.

The reliability expected for PAM can again be estimated by SNR, since the noise samples N_m in (22) are jointly Gaussian[9]. When (21) is satisfied, they are independent with variance $N_0 \cdot \varepsilon_h^2$, and the SNR in (22) is

[8] While it seems straightforward, dealing with ISI is complicated by at least a couple of factors. One is the amplification or enhancement of the noise component when simply putting the received signal through an equalizer that compensates for the channel. Another is estimating the channel dispersion characteristics accurately at the receiver, especially in the light of the ignorance of the data symbols at the receiver.

[9] A more refined estimate of reliability relates the minimum distance of points in the signal constellation to the noise sample variance. More on this later.

$$
\begin{aligned}
\mathrm{SNR} &= \frac{E|A_k|^2 \cdot \varepsilon_h^4}{N_0 \cdot \varepsilon_h^2} \\
&= \frac{\varepsilon_a^2 \varepsilon_h^2}{N_0}
\end{aligned}
\tag{24}
$$

From a system standpoint, the power of the continuous-time PAM signal (20) (rather than the energy of each pulse) is often important. When (21) is satisfied, the signal power P is easily seen to be ε_h^2/T times the average magnitude-square of the data symbols ε_a^2. Expressed in terms of P, the SNR of (24) becomes

$$
\mathrm{SNR} = \frac{P \cdot T}{N_0} . \tag{25}
$$

For constant SNR, increasing the bit rate by reducing T requires a commensurate increase in power P. For example, doubling the bit rate requires doubling the transmit power.

An interesting question is whether "double power is needed to double bit rate" is a fundamental tradeoff, or simply a characteristic of PAM? The answer is that it is *not* fundamental, and we can do considerably better than PAM. The fundamental limits, as well as ways to approach them, will be addressed shortly.

3.3 Probability of error

The SNR is a rather crude measure of reliability, especially for a more complicated scheme like PAM. For example, if we keep ε_a^2 constant, even while increasing K, the number of points in the constellation, the SNR stays fixed. However, increasing K pays a penalty in reliability for ε_a^2 fixed because the constellation points are shoved closer together, resulting in a greater chance for confusion in the presence of noise.

A more refined measure of reliability than SNR is the *probability of error*. There is more than one definition of probability of error, and the appropriate one depends on circumstances. One alternative is the probability of *symbol error*, which is the probability that a different symbol is detected from the one that is transmitted. Another is the probability of *bit error*, which asks the relative frequency with which the bits that are mapped into symbols are in error. The relationship between bit errors and symbol errors is complicated because it depends on how bits are mapped into symbols. In the context of channel coding discussed later, the appropriate measure of reliability is the probability that one multi-dimensional code word is substituted for another.

Consider the probability of symbol error. It is usually much easier to calculate the probability that a symbol is detected correctly. For a constellation like that of Figure 1, a symbol is detected correctly if the noise is smaller than half the distance to the nearest incorrect constellation points. Because the probability distribution function of Gaussian noise is very steep, especially at high SNR the probability of correct detection is therefore dominated by the distance to the *nearest* incorrect constellation point. Thus, it is an accurate approximation to assume that the probability of symbol error is dominated by the *minimum distance* between signal constellation points[10]. The design of signal constellations can therefore focus on achieving the largest minimum distance possible consistent with a given ε_a^2 and K.

The important concept of minimum distance generalizes to more complicated schemes, such as the channel coding considered later. Generally speaking, in the presence of white Gaussian noise, signal design focuses on achieving the largest minimum distance (in the Euclidean sense), whether that be in two dimensions (complex-valued signal constellations) or in higher dimensional spaces (for channel coding, as discussed later).

3.4 Spread spectrum

Another characteristic of interest for a digital communication technique is *spectral efficiency* ν, defined as the bit rate per unit of signal bandwidth (with units of bits per second per Hz). Especially for radio communication, where a fixed and finite spectrum must be divided among many uses, spectrum is quite valuable and there is a motivation to conserve it. This implies maximum spectral efficiency.

The bandwidth of signal (20) is completely determined by the bandwidth of pulse $h(t)$. This in turn is constrained by orthogonality condition (21). It is basically a trivial restatement of the sampling theorem to say that the minimum bandwidth required to satisfy (21) is $W \geq 1/2T$, and in practice $W > 1/2T$ is required since ideal lowpass filters cannot be constructed. Assuming that m bits are communicated with each symbol, the spectral efficiency thus satisfies the constraint that

$$\nu = \frac{m/T}{W} = \frac{m}{WT} \leq 2m\,. \tag{26}$$

Thus, the larger the signal constellation the greater the spectral efficiency.

What if we increase W by choosing a pulse $h(t)$ with larger bandwidth than $1/2T$, in fact *much* larger? First, this does not by itself affect the reliability with coherent detection, since the SNR at the output of the matched filter depends on ε_h^2 and not bandwidth. In practical terms this does make it easier to satisfy (21), but this is a minor advantage. Based on (26) this adversely impacts the spectral efficiency. Particularly in view of this,

[10]This can be formally shown using the union bound of probability. The probability of symbol error is the union of a set of events, one corresponding to the probability that one other constellation point is mistaken for the correct one. The union bound then states that the probability of a union of events is less than the sum of the probabilities of the individual events. This sum is then dominated by the nearest points.

it is perhaps surprising that many radio communication systems today are designed with a bandwidth much larger than the minimum. These systems are said to be *spread spectrum* [4, 5]. Why? Here are some reasons:

- First, do no harm. Increasing bandwidth does not affect the reliability of communication in white Gaussian noise[11].
- Most terrestrial radio communication systems have to deal with *interference* from other systems simultaneously using the same bandwidth. For example, in cellular telephone systems the same bandwidth is reused in different cells, so that the only separation between uses is spatial rather than time or frequency. As another example, WiFi shares a common spectrum with cordless phones, microwave ovens, etc. Interference is often a much more severe impairment than thermal noise or the background radiation of the universe (both well modeled as white Gaussian noise). Dramatically increasing the bandwidth turns out to be very helpful in dealing with interference. There are two complementary benefits. First, on the transmit side, our transmissions may create less interference into other systems if the transmitted power spectrum (transmitted power per unit of bandwidth) is smaller. For a fixed total transmitted power (and hence energy per symbol), increasing the bandwidth can reduce the transmit power spectrum (if the signals are designed with that in mind). Second, on the receive side if an interferer has less bandwidth than our own received signal, the matched filter will attenuate that interference signal more than our received signal.
- Spread spectrum can achieve greater immunity to multipath distortion in radio channels [6].
- Our spectral efficiency calculation took into account only a single point-to-point usage of a given block of spectrum. In fact, spread spectrum is an interesting way to achieve *multiple-access* communication, where multiple users are sharing a common spectrum simultaneously. The basic idea is to keep different uses that are sharing a common spectrum from interfering with one another by allowing each to choose a different waveform $h(t)$, those waveforms chosen to be mutually orthogonal, so a filter matched to one does not respond to the others. This becomes easier as W increases. This way of separating communications (as opposed to relegating them to different time or frequency) has compelling advantages, particularly where data sources are highly bursty (as in internet access, and to a lesser extent speech).

Having said this, designers of non-radio systems using cable are single-access and typically try to conserve bandwidth as much as possible.

[11] It does decrease the SNR at the receiver *input*, but a matched filter detector removes enough of this noise that post-filtering the SNR is bandwidth-independent.

4 Fundamental limits

Modern electronic communications systems are all-digital all the time. The genesis of this technology was during World War II, where the motivation was secrecy. At the end of the war, based in part on the wartime work, Bernie Oliver, John Pierce, and Claude Shannon (all of Bell Laboratories) published an influential paper that laid out the technical arguments for digital as opposed to analog communication [7] and described the basic tradeoffs[12].

At nearly the same time, Shannon published another paper [8] that revolutionized our conception of communications, a work that was grounded in research on cryptography during the war. In some ways Shannon's impact on communications is analogous to Einstein's Special Theory of Relativity. Like Einstein's observation that energy or information cannot be conveyed at speeds higher than c, the speed of light, Shannon proved that communication in the presence of noise cannot exceed C bits per second, where C is the *channel capacity*. The major difference is that C depends in a complicated way on the channel model that Shannon was able to quantify. Like special relativity, where this starting point led to the famous equivalence of mass and energy expressed as $E = mc^2$, Shannon's developed a number of other fundamental insights. These included formal tradeoffs and limits in representing analog sources digitally. Shannon proved that in a fundamental sense nothing is lost by digitally communicating analog sources. Collectively, this work came to known as the *information theory* [9]. Much like special relativity, information theory quickly became famous and spawned many proposals for applicability to other fields, some of which turned out to be intellectual dead ends. The impact on communications was profound, and surely Shannon can be called the father of digital communications.

We will illustrate Shannon's ideas with some intuitive reasoning and simple examples. We start by considering a seemingly different problem, that of the coding of a source into a digital bit stream. The reason for this seeming diversion is that some key concepts are illustrated simply and clearly, and later easily carry over to understanding the problem of communicating a bit stream through a channel.

4.1 Source coding

We begin with the challenge of representing information digitally. We start with the simple situation of representing a sequence of independent coin tosses, and later extend this to outcomes of a Gaussian random process.

[12]Oliver later became the chief technical officer of HP and an early and major proponent of SETI and John Pierce later became the inventor and proponent of satellite-based communications.

Binary representation of coin tosses

Consider a sequence of independent coin tosses, where the probability of a HEAD is p (and hence the probability of a TAIL is $1-p$). When $p = 0.5$ the coin is said to be *fair*. Our goal is to represent this sequence of coin tosses by a sequence of bits, a process called *source coding*. A measure of the coding efficiency is the *rate* of the source code, defined as the average number of bits per coin toss in the digital representation. The reason we have to include the word "average" is that the best coding techniques (best in terms of minimizing the rate) will be stochastic, exploiting any statistical unfairness of the coin.

Example. The simplest source code represents each toss by a single bit, "0" for a TAIL and "1" for a HEAD, as shown in the following table.

Outcome	Code word	Probability
TAIL	0	$1-p$
HEAD	1	p

This straightforward code has a rate of one bit per toss. It has the advantage that the number of bits in the representation is transparent to the actual sequence of coin tosses, but it is also inefficient for the same reason.

Assume the coin is unfair and we know $p \neq \frac{1}{2}$. Then a lower rate can be achieved, as illustrated by the following example.

Example. Assuming that $p > 0.5$ then we expect statistically successive HEAD's in a row more frequently than in the fair case. The *run-length* code of the following table exploits this observation by using n-bit code words to represent the number of HEAD's in a row, rather than to represent individual HEADS or TAILS directly.

Outcome	Code word	Probability
T	000	$1-p$
H T	001	$(1-p)\cdot p$
H H T	010	$(1-p)\cdot p^2$
H H H T	011	$(1-p)\cdot p^3$
H H H H T	100	$(1-p)\cdot p^4$
H H H H H T	101	$(1-p)\cdot p^5$
H H H H H H T	110	$(1-p)\cdot p^6$
H H H H H H H	111	p^7

Note that the code words are fixed in length at three bits, but each generated code word

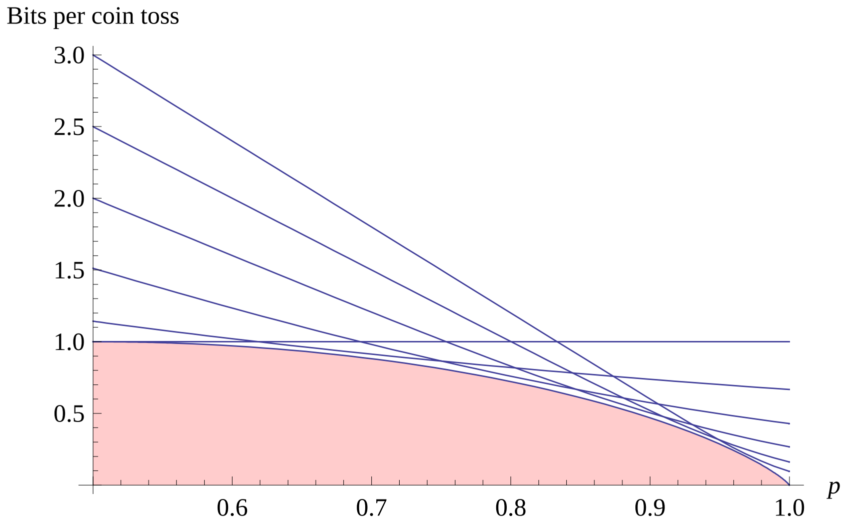

Figure 2: The rate of a run-length code for codeworld size of 1 through 6 bits plotted against the probability of a HEAD. The shaded region is prohibited for *any* source code.

consumes a variable number of tosses[13].

More generally, letting n be the number of bits per codeword ($n = 3$ for the table) and $N = 2^n$, the average number of tosses consumed for each generated n-bit code word is

$$\text{Average tosses} = \sum_{k=1}^{N-1} k \cdot (1-p) \cdot p^{k-1} + (N-1) \cdot p^{N-1} = \frac{1-p^{N-1}}{1-p} \tag{27}$$

and the rate of the code is n divided by this average. The rate is plotted in Figure 2. The reason this code has rate less than unity for $p >> 0.5$ is the small number of bits generated for longer runs of HEADs. In particular, for the code in the table, four through seven heads in a row are represented by only three bits.

Lower bound on rate

Recall that the rate of a code is the average number of bits generated per coin toss. Let R denote this rate. Shown in Figure 2 is a shaded region that cannot be attained by any source coder, no matter how complicated or sophisticated. The attainable region is defined by

[13]This is called a fixed-length code. It is also straightforward to use a variable-length code that generates a variable-length codeword corresponding to a fixed number of tosses.

$$
\begin{aligned}
&R \geq H(p) \\
&H(p) \equiv -p \cdot \log_2 p - (1-p) \cdot \log_2(1-p)
\end{aligned} \tag{28}
$$

Physicists will recognize the $H(p)$ as a familiar formula for entropy.

Short of a formal proof, we will construct a simple argument for (28), one that provides considerable insight. Assume that we observe n independent coin tosses, and devote the resulting vector by $\mathbf{x} = [x_1, x_2, ..., x_n]$. If this vector contains m heads and $n - m$ tails, then due to the independence of the coin tosses the probability of this sequence is

$$p_{\mathbf{X}}(\mathbf{x}) = p^m \cdot (1-p)^{n-m} \,. \tag{29}$$

By the law of large numbers, as $n \to \infty$ we can divide all 2^n possible coin toss sequences into two camps, the relatively high-probability sequences in Ω_h and the relatively low-probability sequences in Ω_h^c, the complement of Ω_h. The high-probability sequences ($\mathbf{x} \in \Omega_h$) have a relative frequency of HEADs that is predicted by the probability p; that is, have the property that $m \approx n \cdot p$. What is the probability of a typical $\mathbf{x} \in \Omega_h$? Substituting $m = n \cdot p$ in (29) we get

$$p_{\mathbf{X}}(\mathbf{x} \in \Omega_h) \approx p^{np} \cdot (1-p)^{n(1-p)} = 2^{-nH(p)} \,. \tag{30}$$

A careful application of the law of large numbers confirms that the probability of outcomes in Ω_h approach unity as $n \to \infty$. This implies that the probability of outcomes in Ω_h^c approach zero. Further, the probability in (30) is constant, independent of $\mathbf{x} \in \Omega_h$. Since these sequences in Ω_h have the same probability asymptotically, and their probabilities sum to unity, the number of sequences in Ω_h must be the reciprocal of their probability, or asymptotically $2^{nH(p)}$.

Based on this insight, we can define a feasible coding scheme that works asymptotically and requires only $H(p)$ bits to represent each coin toss, or $nH(p)$ bits to represent n tosses. With $nH(p)$ bits we can represent $2^{nH(p)}$ sequences, so choose to represent only those high-probability sequences in Ω_h. This is efficient because all these sequences have asymptotically the same probability. We have no remaining bits to represent the low-probability sequences in Ω_h^c, so if one of those sequences comes along we will be in deep trouble. Fortunately we don't have to obsess over this problem for large n since the probability of Ω_h^c approaches zero.

This code's effectiveness depends on the asymptotic properties predicted by the law of large numbers as $n \to \infty$, and thus any feasible code for finite n would not be able to do quite as well in terms of rate. Hence it credible that this represents a lower bound on the feasible rate of any finite-n code.

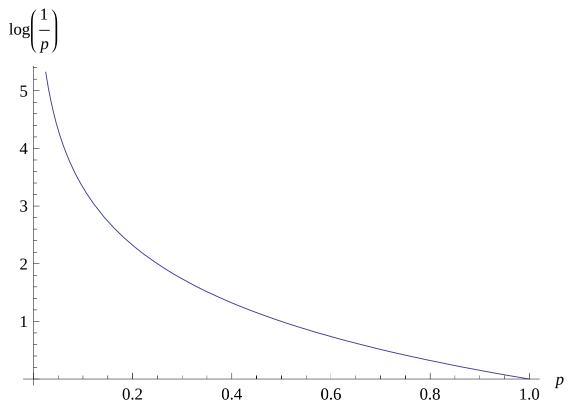

Figure 3: The uncertainty of a given outcome $X = x$ of a random variable X vs. its probability.

Both the run-length code and the lower bound argument depended on certain statistical properties of sequences of coin tosses. The key conceptual tool was abandoning the independent coding of each coin toss, and instead mapping sequences of coin tosses into code words with multiple bits.

Uncertainty of a random variable

The results of the last section are easily generalized. Assume we are given a random variable X with probability mass function $p_X(x_i)$. For a specific outcome $X = x_i$, define the *uncertainty* of X as $\log_2(1/p_X(x_i))$, with the units of bits. This function of probability is plotted in Figure 3. Intuitively this quantity is a measure of the prior uncertainty of $X = x_i$ before it is observed. When the probability of $X = x_i$ is unity, then x is the certain outcome and as a result the uncertainty is zero. When outcome $X = x_i$ has probability $1/2$, its uncertainty is one bit, probability $1/4$ makes its uncertainty two bits, probability $1/8$ makes its uncertainty three bits, etc. As the probability approaches zero, the uncertainty approaches infinity.

In a sense, uncertainty is a measure of significance or surprise associated with a particular outcome. If some outcome is a priori very improbable, then its occurrence is more surprising and thus conveys more information. For example, a ”one” coming up on a six sided die carries more information ($\log_2 6 = 2.58$ bits) than a ”head” turning up in a coin toss ($\log_2 2 = 1.0$ bits), simply because it is less expected and more surprising.

Performing a weighted average of the uncertainty over all outcomes of X, weighted by the probability of that outcome, the result is the *average uncertainty* given by

$$
\begin{aligned}
H(X) &= E_X[-\log_2 p_X(X)] \\
&= -\sum_{k=1}^{K} p_X(x_k) \log_2 p_X(x_k)
\end{aligned} \tag{31}
$$

Here the E_X operator is expectation (ensemble average) over the ensemble of random variable X (see Appendix A for notational conventions). The use of the expectation operator in (31) is sometimes confusing to readers. The idea is, given a random variable X, to define a new random variable $Y = -\log_2 p_X(X)$. Y has the interpretation as the uncertainty of X for each possible outcome of X. The average uncertainty is then the average of Y over all possible outcomes of X weighted by the probability of those outcomes. The probability mass function thus appears in two different roles, first as a predictor of the uncertainty of each outcome, and second as the factor weighting those uncertainties by the probabilities of each outcomes.

$H(X)$ is the average uncertainty in X before observation, or equivalently the average information obtained in the process of observing X. If for example we observe a sequence of statistically independent and identically distributed random variables each with probability mass or density function $p_X(x)$ and average uncertainty $H(X)$, and encode these outcomes digitally using bits, then the average rate R (expressed as bits per observation) must satisfy $R \geq H(X)$.

Example (unfair coin toss). Figure 4 plots the average uncertainty of an unfair coin toss,

$$
H(X) = -p \log_2 p - (1-p) \log_2 (1-p) \,. \tag{32}
$$

Three observations are in order:

- Although the uncertainty of a specific outcome may be very high, by definition that outcome is improbable and thus does not affect the *average* uncertainty to the same degree that we might expect. In particular, when $p \approx 1$ a TAIL has a very high uncertainty and yet the average uncertainty is very low because the low-uncertainty HEAD is much more probable.
- The average uncertainty is symmetrical in p; either $p \approx 0$ or $p \approx 1$ result in low average uncertainty. Reversing the labeling of HEADs and TAILs will not affect the average uncertainty.
- The actual value of the random variable X does not enter into the uncertainty. The coin toss outcome might be represented by "0" and "1" rather than "HEAD" and "TAIL" without affecting the uncertainty or average uncertainty.

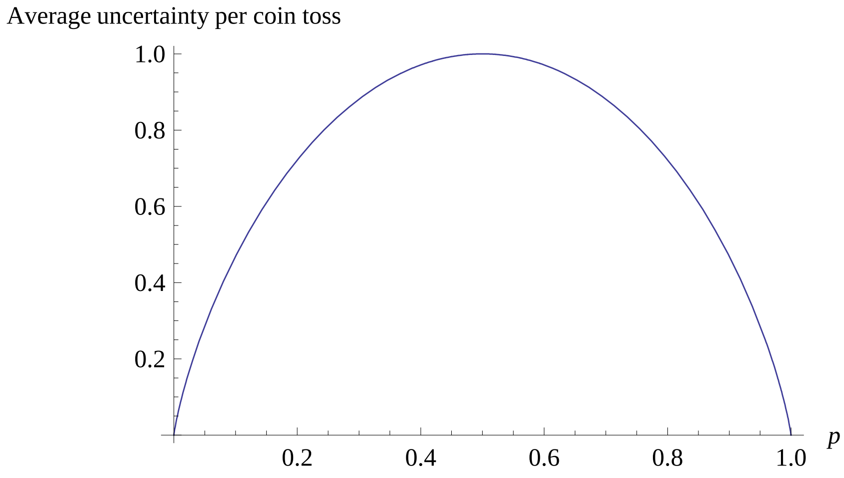

Figure 4: $H(X)$ for an unfair coin toss with p the probability of a HEAD.

The formula for average uncertainty mirrors the entropy that arises in thermodynamics, and for that reason $H(X)$ is often called *information entropy.* Uncertainty in this communication context lacks the physical significance attached to entropy in thermodynamics, so information uncertainty is not equivalent to thermodynamic entropy in any sense more profound that sharing a common formula[14].

Maximum uncertainty

What is the largest possible value for average uncertainty $H(X)$? Random variables for which outcomes are closer to equally likely have higher average uncertainty, as illustrated by the coin toss example. It is shown in Appendix B that the average uncertainty as defined by (31) is bounded by

$$0 \leq H(X) \leq \log_2 K \tag{33}$$

with equality in the upper bound if and only if the outcomes are equally likely, or $p_X(x_k) = 1/K$ for $1 \leq k \leq K$.

[14]Physicists will recognize an argument parallel to our coin tossing example from thermodynamics. In a system at thermal equilibrium, if there are particles with two energies, then these particles occur with a predictable relative frequency. The state of the system is thus analogous to coin toss sequences, and similar law of large number arguments lead to an entropy formulation as the number of particles approaches infinity.

Average uncertainty for continuous random variables

The formula for (31) can formally be extended to discrete random variables with a countable infinity of outcomes ($K \to \infty$) and continuous random variables. In the latter case, we can define

$$\begin{aligned} H(X) &\doteq E_X[-\log_2 p_X(X)] \\ &= -\int p_X(x) \log_2 p_X(x)\,. \end{aligned} \tag{34}$$

While this definition is very useful, it does not have the interpretation as average uncertainty. For this reason, it is given a different name, *differential entropy*[15]. Some mathematical difficulties with this definition can be illustrated by applying it to a Gaussian random variable.

Example. Let X be a Gaussian random variable with mean μ and variance σ^2. Then

$$\begin{aligned} -\log_2 p_X(x) &= -\log_2 \left[\frac{1}{\sigma\sqrt{2\pi}} \cdot e^{-(x-\mu)^2/2\sigma^2} \right] \\ &= \log_2[\sqrt{2\pi} \cdot \sigma] + \frac{(x-\mu)^2}{2\sigma^2} \log_2 e \end{aligned} \tag{35}$$

This is plotted in 5 for some different values of standard deviation σ.

$H(X)$ is obtained by substituting X for x and taking the ensemble average E_X,

$$\begin{aligned} H(X) &= E_X \left[\log_2[\sqrt{2\pi} \cdot \sigma] + \frac{(X-\mu)^2}{2\sigma^2} \log_2 e \right] \\ &= \log_2[\sigma\sqrt{2\pi e}] \end{aligned} \tag{36}$$

This is plotted in Figure 6. $H(X)$ increases with increasing standard deviation, but does not depend on μ.

One surprising property of $H(X)$ in (36) is that $H(X) < 0$ for $\sigma^2 < 2\pi e$. The problem is that for a continuous random variable it is possible to have $p_X(x) > 1$, which can in turn cause $H(X) < 0$. Thus, since it makes no sense to say that the rate of a source

[15] The mathematical details are covered in Chapter 9 of [9].

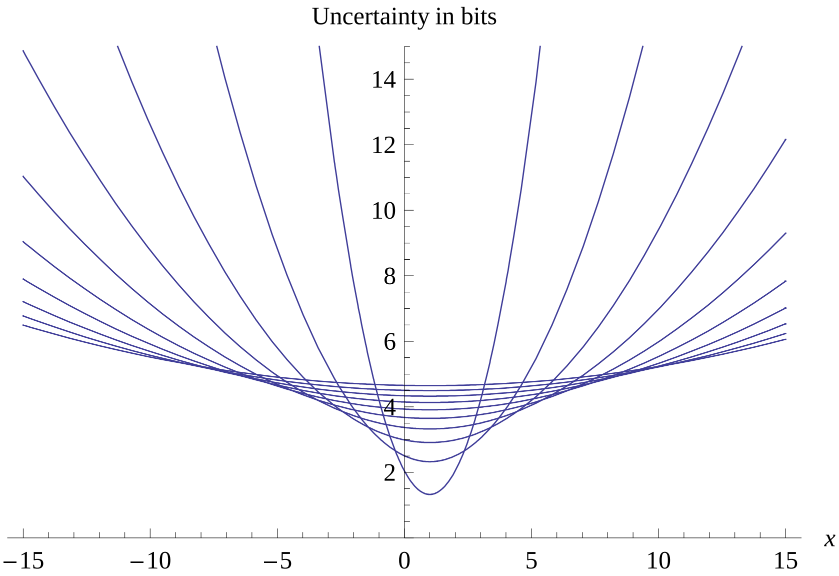

Figure 5: The value of $-\log_2 p_X(x)$ for a Gaussian random variable with mean $\mu = 1$ and standard deviation $\sigma = 1$ through 10 in steps of 1. The more eccentric curves are smaller values of σ.

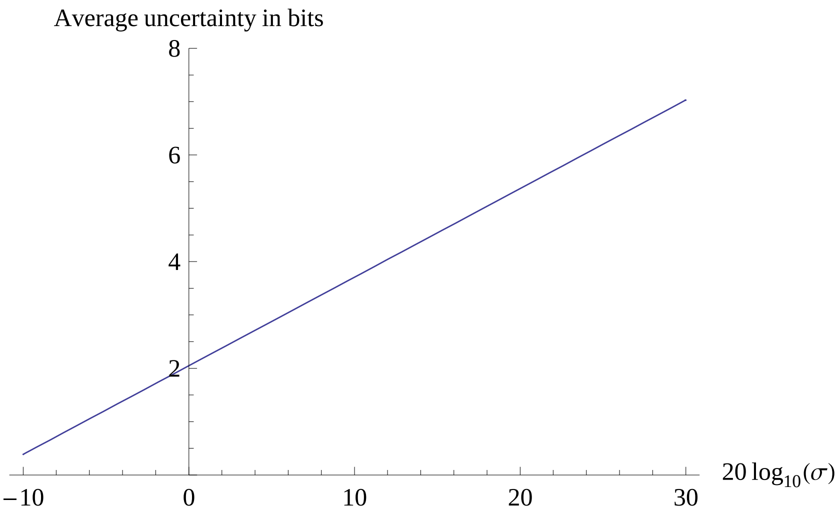

Figure 6: $H(X)$ for a Gaussian random variable vs the standard deviation in dB. The uncertainty actually goes negative for very small deviations, illustrating a mathematical difficulty with uncertainty for continuous random variables.

coder can be negative, $H(X)$ certainly does not have the same interpretation as in the discrete case.

Another disturbing property of (36) is that $H(X)$ depends on the units of σ. This is another indication of problems in interpreting $H(X)$ in terms of something physically meaningful like source coding rate.

A third property of (36), but one that is not so surprising, is that $H(X) \to \infty$ as $\sigma \to \infty$. Generally speaking, when working with $H(X)$ it is necessary to place constraints, such as maximum variance, lest $H(X)$ becomes unbounded.

In spite of the difficulties in definition (34), it turns out to be very useful. $H(X)$ is not a meaningful quantity in itself, but it is very powerful as a means to an end. That end is calculating the channel capacity.

It is shown in Appendix B that if the variance of a continuous zero-mean random variable X is constrained to $E_X[X^2] \leq \sigma_X^2$, then

$$H(X) \leq \log_2 \left[\sigma\sqrt{2\pi e}\right] \tag{37}$$

with equality in the upper bound if and only if X is Gaussian. In words, a Gaussian random variable has the largest $H(X)$ among all variance-constrained continuous random variables. This fact will prove useful in calculating the capacity of a Gaussian noise channel, as will be pursued next.

4.2 Channel capacity

Of great interest is the maximum bit rate that can be reliably communicated through a given noisy channel. This can be illustrated for the discrete-time channel of (3) where $g_k = \delta_K$ and the N_k are statistically independent zero-mean Gaussian random variables. This channel is then characterized by the model

$$Y_k = X_k + N_k\,. \tag{38}$$

For simplicity assume that these variables are all real-valued. We call $X_k \to Y_k$ a single *channel use*, and calculate the capacity in bits per channel use.

Recall that in (20) we displayed a specific signalling scheme called PAM where $X_k = A_k$ and A_k is chosen from a finite alphabet. The question now addressed is whether PAM falls short of fundamental limits, the channel capacity, and if so by how much. The answer is that it does fall short, and by pretty far. We then illustrate how channel coding can push closer to those limits.

Unless the input signal level is constrained, the channel capacity is infinite, so it is appropriate to introduce a power constraint on the channel input signal,

$$E_X[X^2] \le \sigma_X^2 \,. \tag{39}$$

For this power-constrained channel, the *channel capacity* is

$$C = \frac{1}{2}\log_2\left[1 + \frac{\sigma_X^2}{\sigma_N^2}\right] \,. \tag{40}$$

This will be derived shortly, but first let's explore the implications of this result.

The interpretation of C comes from a channel coding theorem, which states roughly the following. If we attempt to communicate a stream of bits with average rate R bits per channel use, and if $R > C$, then the error rate is bounded away from zero no matter how complex or sophisticated the techniques used. That is, the channel is intrinsically unreliable at these rates. On the other hand, if $R \le C$ then there exist transmitter-receiver designs for which the error rate approaches zero asymptotically as the latency approaches infinity[16]. By latency, we mean the delay from input bit stream to output bit stream, the cause of which is the statistical averaging necessary to achieve reliability. In other words, at rates below capacity it is possible to use the channel for reliable communication.

Figure 7 plots the capacity of (40) against the SNR in decibels. The capacity increases as the SNR increases, achieving an impressively large number of bits per channel use at high SNR's.

The channel coding theorem unfortunately is not constructive; that is, it states the possibilities without actually telling us precisely how to achieve them. Actually trying to achieve reliable communication at rates near capacity has turned out to a difficult challenge due to an exponential growth in computational complexity as the latency and reliability grows. Communications practice has seen a gradual improvement in the rates achievable due to the march of Moore's law coupled with conceptual breakthroughs. Finally, after about five decades of research and technology advancement, it is possible to achieve rates close to capacity in practice.

Before Shannon's extraordinary contribution in 1948, two somewhat contradictory assumptions persisted. Practitioners generally were optimistic that communication throughput and reliability could always be improved with ever more sophisticated techniques, while theorists usually assumed it would be unthinkable to communicate reliably over

[16]The proof is rather long and involved and is omitted. However, the basic idea behind proving the $R < C$ result is as simple as it is clever. A random ensemble of channel codes is defined (channel codes will be addressed later), and the error probability averaged over that ensemble is shown to approach zero at rates below capacity. At least one code in the ensemble must match or exceed the average. This proof is not constructive and thus gives little hint as to how capacity can be achieved in practice.

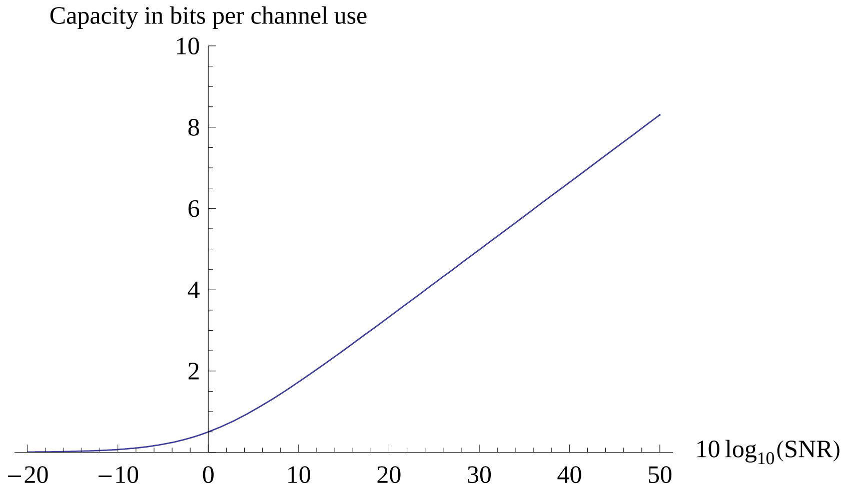

Figure 7: The channel capacity per channel use in bits for an additive Gaussian noise channel with independent noise samples with variance σ_N^2 and channel input variance constrained to σ_X^2. The SNR is defined as σ_X^2/σ_N^2.

a noisy channel (noise is synonymous with unreliability). Shannon showed that both groups were technically incorrect, in the sense that arbitrarily reliable communication is possible (at the expense of unlimited latency) but only as long as the rate is below the channel capacity, and at rates above the channel capacity unreliability is inevitable no matter how complex or sophisticated the technology. Shannon also firmly established theory (as opposed to experimentation and implementation) at the forefront of future practical advances in communications.

We now illustrate in more detail where the capacity formula comes from. Intuitively, at least, this is a straightforward extension of the source coding results presented earlier.

Equivocation

In the simple memoryless single-input single-output case any channel model can be characterized statistically by the conditional probability mass or density function $p_{Y|X}(y|x)$. This specifies, for each channel input, the distribution of the channel output. The probability mass or density of the input $p_X(x)$ is determined not by the channel model but by our transmitter design and source statistics. Together $p_X(x)$ and $p_{Y|X}(y|x)$ determine the joint input-output probability mass or density function through

$$p_{X,Y}(x, y) = p_{Y|X}(y|x) \cdot p_X(x) \,. \tag{41}$$

From this joint mass or density we can determine other quantities of interest, such as

$p_{X|Y}(x|y)$ and $p_Y(y)$.

Suppose we observe the channel input $X = x$. Conditional on this observation, the remaining uncertainty about output random variable Y is $\log_2[1/p_{Y|X}(y|x)]$. As in the case of a single random variable, this uncertainty can be averaged over the (X, Y) ensemble, as in

$$\begin{aligned} H(Y|X) &\doteq E_{X,Y}\left[-\log_2 p_{Y|X}(Y|X)\right] \\ &= -\sum_{k=1}^{K}\sum_{m=1}^{M} p_{X,Y}(x_k, y_m) \cdot \log_2 p_{Y|X}(y_m|x_k)\,. \end{aligned} \tag{42}$$

Note that the random variables X and Y have been substituted for the arguments x and y of $p_{Y|X}(y|x)$, and the result has been averaged over all outcomes of (X, Y). The result is the uncertainty that is resolved in observing $Y = y$ even after being told the channel input $X = x$, averaged over not only all Y but also over all channel inputs X. $H(Y|X)$ of (42) is called the channel *equivocation*.

Example. Our interest is in the Gaussian channel of (38). Formally, the definition of (42) can be applied to a continuous channel input and output, although this is not without its perils. This turns out to be a useful intermediate step to determining the capacity, although the interpretation as an average uncertainty is not valid.

For our example, since $Y = X + N$ we know that conditional on $X = x$, $Y = x + N$ is Gaussian with mean value x and variance σ_N^2, so the uncertainty of $Y = y$ is

$$\log_2[\sigma_N\sqrt{2\pi}] + \frac{(y-x)^2}{2\sigma_N^2}\log_2 e \tag{43}$$

The next step is to average this quantity over X and Y. First noting that

$$E_{X,Y}(Y-X)^2 = E_N N^2 = \sigma_N^2 \tag{44}$$

we get

$$\begin{aligned} H(Y|X) &= E_N\left[\log_2[\sigma_N\sqrt{2\pi}] + \frac{N^2}{2\sigma_N^2}\log_2 e\right] \\ &= \log_2 \sigma_N\sqrt{2\pi e}\,. \end{aligned} \tag{45}$$

Actually, in this case the equivocation is $H(Y|X) = H(N)$, the differential entropy of the noise. This makes intuitive sense, since the additive Gaussian noise is assumed independent of the channel input.

Transinformation

The *transinformation* (also called the *mutual information*) is defined as

$$\begin{aligned} I(X,Y) &\doteq E_{X,Y}\left[\log_2 \frac{p_{X,Y}(X,Y)}{p_X(X)\cdot p_Y(Y)}\right] \\ &= \sum_{k=1}^{K}\sum_{m=1}^{M} p_{X,Y}(x_k,y_m)\cdot \log_2 \frac{p_{X,Y}(x_k,y_m)}{p_X(x_k)\cdot p_Y(y_m)}\,. \end{aligned} \tag{46}$$

The quantity $I(X,Y)$ as defined in (46) is, it turns out, perfectly legitimate in the case of continuous as well as discrete random variables. It can't do weird things like go negative, and it is not dependent on the units with which parameters are specified.

For purposes of interpretation, it is straightforward to rewrite $I(X,Y)$ in two different forms,

$$\begin{aligned} I(X,Y) &= H(Y) - H(Y|X) \\ &= H(X) - H(X|Y)\,. \end{aligned} \tag{47}$$

The interpretation is easier for the second form, since our goal is to infer as much information about the channel input as possible from observing the channel output. At least in the discrete case, $H(X)$ is the average uncertainty about X without observing the channel output, and the equivocation $H(X|Y)$ is the average uncertainty of X conditional on observing the channel output $Y = y$. The transinformation is a measure of how much the average uncertainty of X is *reduced* by observing the channel output, or in other words how much uncertainty about X is resolved by one channel use.

Example. When the channel input and output are statistically independent, $p_{X,Y}(x,y) = p_X(x)\cdot p_Y(y)$ and $I(X,Y) = 0$. This indicates that the channel output resolves none of the uncertainty in the channel input, as expected.

The first form of (47) is usually the easiest for calculations, as illustrated by the following example.

Example. For the additive Gaussian noise channel of (38), from the previous example

$$\begin{aligned} I(X, Y) &= H(Y) - H(N) \\ &= H(Y) - \log_2 \sigma_N \sqrt{2\pi e}\,. \end{aligned} \tag{48}$$

The quantities $H(X)$ and $H(N)$, even though they suffer mathematical difficulties in this continuous case, are nevertheless useful as an intermediate step to calculating $I(X, Y)$. For example, if the units of σ_N are changed, both terms in (48) will be affected in such a way that the difference (and hence $I(X, Y)$) will be unaffected.

As this last example illustrates, since the distribution of Y depends on the distribution of X, $I(X, Y)$ cannot be calculated in full until the distribution of X is known. But $p_X(x)$ is not a feature of the channel, but rather of what we choose to feed to the input of the channel. Choosing $p_X(x)$ is the final step in determining the channel capacity.

Channel capacity

We are finally prepared to define the *channel capacity*, which is the largest possible mutual information,

$$C = \sup I(X, Y)\,. \tag{49}$$

This maximizes the information conveyed about X obtained in observing Y, on average. The free parameter in the supremum is the input distribution $p_X(x)$, which can be controlled by our transmitter design.

Example. For the additive Gaussian noise channel of (38), the maximization should also be constrained by the variance (power) of X, since otherwise the capacity will be infinite,

$$E_X[X^2] \le \sigma_X^2\,. \tag{50}$$

From (48), the equivocation $H(Y|X) = H(N)$ is independent of the choice of $p_X(x)$, and thus only $H(Y)$ need be maximized. The variance of Y is known from (50) and from the assumed independence of X and N,

$$\begin{aligned}\sigma_Y^2 &= E[X^2] + \sigma_N^2 \\ &\le \sigma_X^2 + \sigma_N^2 \,.\end{aligned} \tag{51}$$

As shown in Appendix B and restated in (37), $H(Y)$ is bounded by

$$\begin{aligned}H(Y) &\le \frac{1}{2}\log_2\left[2\pi e \cdot \sigma_Y^2\right] \\ &\le \frac{1}{2}\log_2\left[2\pi e \cdot (\sigma_X^2 + \sigma_N^2)\right]\end{aligned} \tag{52}$$

with equality if and only if Y is Gaussian. Fortunately, we can force Y to be Gaussian if X is chosen to be Gaussian, since the additive noise is Gaussian, so this choice maximizes the transinformation and achieves channel capacity. This confirms the capacity formula of (40), plotted in Figure 7. That the capacity is achieved for a Gaussian X suggests that the transmitter design should seek to make the channel input obey a Gaussian distribution.

Continuous-time channel

Consider the continous-time channel of (1) where[17] $g(t) = \delta(t)$ and $N(t)$ is a wide-sense stationary random process that is Gaussian and white with power spectral density N_0. Let's investigate the capacity of this channel when the input signal $X(t)$ is constrained to have bandwidth W and $E_X[X^2] \le \sigma_X^2$. At the receiver we can pass the channel output through an ideal lowpass filter with bandwidth W, and the sampling theorem allows that the resulting signal can be sampled at twice the bandwidth, or $2W$ samples per second. The resulting discrete-time representation is equivalent in the sense that the continuous-time signal can be exactly reconstructed from it. The resulting channel including filter and sampler conforms to the discrete-time model of (3) with noise variance equal to $N_0 W$ and, since the signal is not affected by the lowpass filter, $E_X[X_k^2] \le \sigma_X^2$. The total capacity is then the capacity per use from (40) times the sampling rate $2W$, or

$$C = W \cdot \log_2\left[1 + \frac{\sigma_X^2}{W \cdot N_0}\right] \text{ bits per second} \tag{53}$$

The bandwidth W enters this formula in two ways:

[17] Convolving with the Dirac delta function $\delta(t)$ does not change the input.

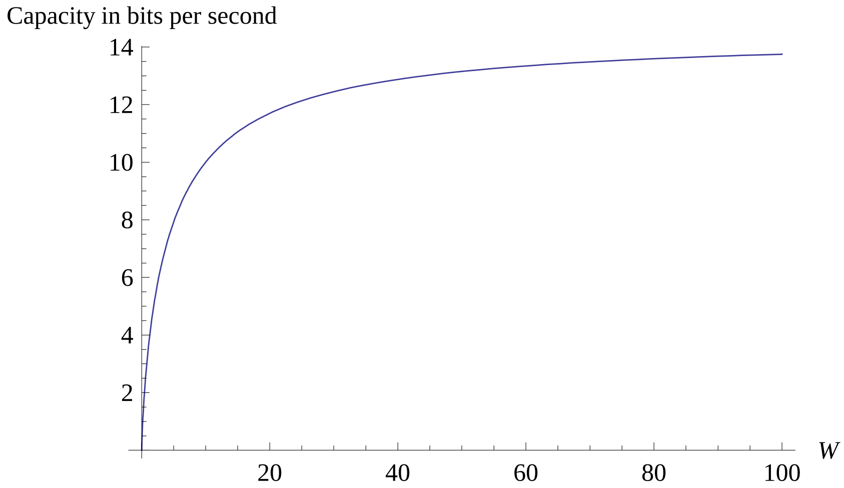

Figure 8: The capacity C of a continuous-time white Gaussian noise channel as a function of its bandwidth for fixed $\sigma_X^2/N_0 = 10$. As bandwidth W increases, the capacity increases in spite of the deterioration in SNR.

- There is a proportional term that drives capacity up linearly with W. Intuitively, this term reflects the ability to transmit more data symbols per unit time if W is larger, thus increasing the symbol rate and the data rate.
- Increasing W also admits more noise, reducing the SNR term, and hence reducing the maximum spectral efficiency C/W (in bits per second per Hz).

Fortunately, the spectral efficiency decreases more slowly than the bandwidth increases, so the net effect is to increase the capacity as the W increases. It is useful to remind ourselves that (53) is a fundamental bound on the bit rate that can be achieved reliably, and does not tell us anything about how to achieve those bit rates reliably[18].

The important point, reiterated in the plot of Figure 8, is that the capacity increases monotonically with bandwidth. This confirms that for an additive white Gaussian noise channel, increasing the bandwidth is always beneficial in the sense of allowing reliable communication at higher bit rates. However, the effect saturates, with further increases in bandwidth having less and less benefit.

It is easily confirmed that the capacity is bounded as $W \to \infty$. Define

$$\gamma = \frac{W \cdot N_0}{\sigma_X^2} \tag{54}$$

in which case (53) can be written as

[18]Any association of terms in this bound with parameters of practical schemes like PAM should be approached with caution.

$$
\begin{aligned}
C &= \frac{\sigma_X^2}{N_0} \cdot \log_2 \left[1 + \frac{1}{\gamma}\right]^{\gamma} \to C_\infty \\
C_\infty &= \frac{\sigma_X^2}{N_0} \cdot \log_2 e = 1.44 \cdot \frac{\sigma_X^2}{N_0} \,.
\end{aligned}
\tag{55}
$$

The larger the ratio of σ_X^2 to the noise spectral density N_0, the larger C_∞, the asymptotic capacity at infinite bandwidth. In other words, if the signal power is allowed to be larger or if the noise power is smaller, C_∞ is increased.

Another insightful way to look at the capacity formula is to define the spectral efficiency ν at capacity and the SNR as

$$
\begin{aligned}
\nu &= \frac{C}{W} \\
\mathrm{SNR} &= \frac{\sigma_X^2}{W \cdot N_0}
\end{aligned}
\tag{56}
$$

where this SNR is the incoherent SNR defined earlier, the ratio of total signal power to total noise power at the receiver *input*. Then the relation between these two quantities at capacity is

$$
\mathrm{SNR} = 2^{\nu} - 1 \,. \tag{57}
$$

The relation is plotted in Figure 9. As expected, the increasing spectral efficiency requires increasing SNR, and at high ν this relationship becomes linear when SNR is expressed in decibels. What may be unexpected is that it is possible to communicate reliably at a large *negative* SNR (where the total noise power is much larger than the signal power), but this requires that $\nu << 1$. For example, if $\nu = 0.1$ (spectral efficiency equal to one-tenth of a bit per second per Hz) it is possible to communicate reliably at an input SNR = 0.071 (minus 8.7 dB).

Example. Suppose the spectral efficiency at capacity is $\nu = C/W$ with bandwidth W. What is C_∞ when σ_x^2 and N_0 are kept constant?[19] It is

$$
C_\infty = W \cdot (2^{\nu} - 1) \cdot \log_2 e \,. \tag{58}
$$

The growth in capacity from bandwidth W to bandwidth infinity can be written as

[19]Note that it is nonsensical to talk about the spectral efficiency at infinite bandwidth.

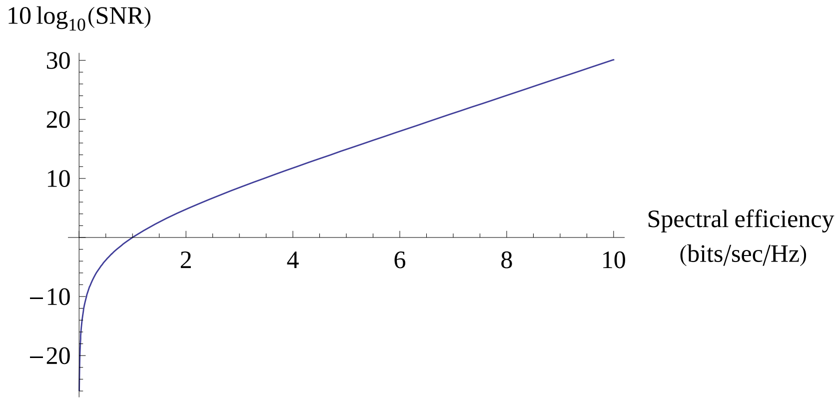

Figure 9: At capacity, the input SNR in dB required to achieve a given spectral efficiency ν.

$$\frac{C_\infty}{C} = \frac{2^\nu - 1}{\nu} \cdot \log_2 e\,. \tag{59}$$

The larger the starting ν (that is, larger SNR), the greater the total *growth* in capacity as bandwidth is increased without bound.

That capacity grows more and more slowly with bandwidth suggests that it might be advantageous to fragment a wide bandwidth channel into a number of smaller bandwidth channels, and use them independently. This is not true (and would violate the channel coding theorem), but the reason is a little subtle.

Example. Suppose bandwidth W is divided into two pieces W_1 and W_2, where $W_1 + W_2 = W$. What is the total capacity of the two sub-channels? A fair comparison requires that the total transmit power σ_X^2 be divided between the two channels, or $\sigma_X^2 = \sigma_1^2 + \sigma_2^2$. The total capacity is then

$$C_1 + C_2 = W_1 \cdot \log_2 \left[1 + \frac{\sigma_1^2}{W_1 \cdot N_0}\right] + W_2 \cdot \log_2 \left[1 + \frac{\sigma_2^2}{W_2 \cdot N_0}\right] \tag{60}$$

A straightforward optimization, maximizing C over σ_1^2 and σ_2^2 subject to the constraints shows that $C_1 + C_2 \leq C$ with equality when power is allocated in proportion to bandwidth, or $\sigma_1^2 = W_1 \sigma_X^2 / W$.

Comparison: PAM vs capacity

The question arises, what gap may or may not exist between the fundamental limit of channel capacity and what can be accomplished with a straightforward technique like PAM. The answer can serve to motivate how much effort should be devoted to more sophisticated channel coding techniques. It turns out that the gap is substantial, and thus efforts devoted to channel coding can pay off handsomely.

For this purpose, let's return to the simple discrete-time channel. Recall that capacity is given by (40) for this case. It is convenient to define a *rate-normalized* SNR, defined as

$$\mathrm{SNR}_{\mathrm{norm}} = \frac{\mathrm{SNR}}{2^{2R} - 1} \tag{61}$$

Then $R \leq C$ is equivalent to $\mathrm{SNR}_{\mathrm{norm}} \geq 1$.

Recall that in PAM we encode R bits of information (R is assumed to be an positive integer) by transmitting a signal $X_k = A_k$ that assumes one of 2^R values. In practice those values will be complex valued as in the constellation of Figure 1, but for simplicity let's chose a real-valued constellation with 2^R equally spaced levels. To minimize the transmit power, space those levels symmetrically about $X_k = 0$, as in $X_k = A_k \in \left[k - \frac{2^R+1}{2},\, 1 \leq k \leq 2^R\right]$ Assuming each of these possibilities is equally likely, the signal variance is easily shown to be

$$\sigma_X^2 = \frac{2^{2R} - 1}{12} \tag{62}$$

and hence the rate-normalized SNR is

$$\mathrm{SNR}_{\mathrm{norm}} = \frac{1}{12\sigma_N^2} \tag{63}$$

Perhaps surprisingly $\mathrm{SNR}_{\mathrm{norm}}$ is independent of R.

The other issue is what level of reliability is actually achieved by PAM? We measure this by the probability of error. Except for the two outlier levels at both ends, a correct decision is made if the noise falls in the range $N \in \{-0.5, 0.5\}$, Assuming Gaussian noise with variance σ_N^2 the probability of error can be approximated as

$$P_e = 1 - \int_{-1/2}^{1/2} \frac{1}{\sqrt{2\pi}\sigma_N}\, e^{-x^2/2\sigma_N^2}\, \mathrm{d}x\,. \tag{64}$$

Rather than plot P_e against SNR, it is convenient to plot P_e vs. $\mathrm{SNR}_{\mathrm{norm}}$ as shown in Figure 10. The region below capacity, where theory tells us arbitrarily reliable operation

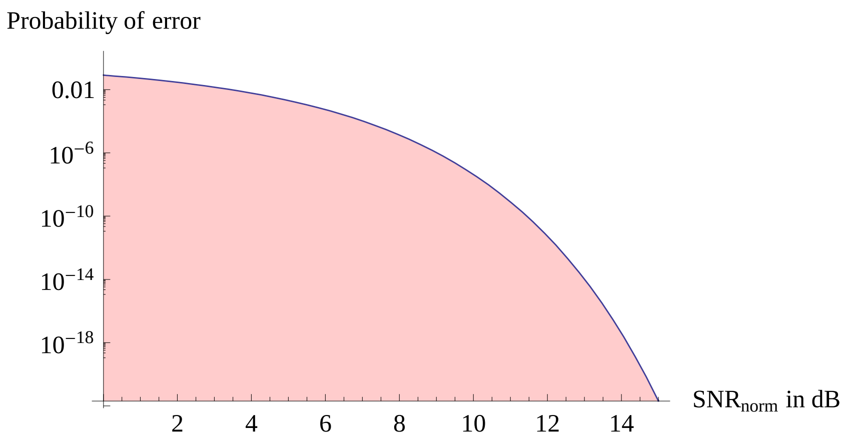

Figure 10: The probability of error plotted against SNR_{norm} in dB. The SNR gap to capacity for a given reliability (P_e) is the distance from the vertical axis ($\text{SNR}_{\text{norm}} = 1$) and the curve at that P_e.

should be possible, is $\text{SNR}_{\text{norm}} \geq 1$, which corresponds to the region to the right of the axis. PAM does operate reliably in this region, but only as SNR_{norm} gets relatively large, considerably larger than demanded by the capacity formula.

The shaded area is the region where the channel coding theorem predicts arbitrarily high reliability is possible at lower SNR's than PAM. Thus, there is considerable opportunity to improve on PAM. In particular, depending on how low a P_e we demand, it should be possible to operate at SNR's on the order of 9 to 11 dB lower than PAM, independent of the rate R that we are attempting to achieve[20]. Thus, the channel coding theorem predicts the ability to operate reliably at noise levels roughly an order of magnitude or so higher in power compared to PAM. This is a substantial improvement, and one that is well worth the effort to capture through channel coding techniques.

Channel coding

How can we actually close the SNR gap to capacity? How can we improve on PAM?

Recall the source coding situation. It was advantageous to code sequences of coin tosses rather than treating them independently. Even in the case where the tosses are statistically independent, for an unfair coin patterns start to emerge in the coin toss sequences in the sense that certain sequences are much more likely than others. For example, if a coin favors HEADs, then a sequence of M HEADs is more likely than a sequence of M TAILs. These properties can be exploited to reduce the average number of bits required

[20]Of course, higher rates require higher SNR's in absolute terms. What we are discussing here is the relative gap in SNR between PAM and capacity. That gap is the same at any rate R.

to represent a coin toss. If tosses are statistically dependent, this becomes even more true.

A similar situation applies to communicating a sequence of bits through a channel. Consider again the example of a discrete-time channel of (38) in which a sequence of input samples is corrupted by independent Gaussian noise samples. Let us investigate how we might increase the reliability of communication on such a channel, beating PAM and reaching closer to channel capacity. The key is to recognize that the unreliability represented in P_e is dominated by "outlier" noise samples that occur occasionally. Instead of treating each channel use independently, as in PAM, we can group sequences of channel uses together and observe (due to the law of large numbers) more consistent sequences of noise samples (in a statistical sense). This is an example of *channel coding*, in which the transmit sequences are tailored to the statistics of noise on the channel. This particular style of channel coding is called a *block* code[21].

Grouping M successive inputs together into a vector-valued random variable $\mathbf{x}$, the resulting channel can be treated as a multiple-input multiple-output (MIMO) channel[22] of the form

$$\mathbf{Y} = \mathbf{x} + \mathbf{N}\,. \tag{65}$$

The M components of vector $\mathbf{N}$ are jointly independent Gaussian random variables, each identically distributed with mean 0 and variance σ_N^2. The MIMO distribution of $\mathbf{Y}$ conditional on $\mathbf{X} = \mathbf{x}$ is

$$p_{\mathbf{Y}|\mathbf{X}}(\mathbf{y}|\mathbf{x}) = \prod_{k=1}^{M} \frac{1}{\sigma_N\sqrt{2\pi}} e^{-(y_k - x_k)^2/2\sigma_N^2} = \frac{1}{\sigma_N^M (2\pi)^{M/2}} e^{-\|\mathbf{y}-\mathbf{x}\|^2/2\sigma_N^2}\,. \tag{66}$$

This probability density is thus expressed in terms of a distance between $\mathbf{x}$ and $\mathbf{y}$ in M-dimensional Euclidean space. The idea in the design of a block channel code is to construct a set of *codewords*, each of which is a point in M-dimensional Euclidean space. While keeping the rate fixed, instead of mapping R bits into a single channel use, $M \cdot R$ bits can be mapped into M channel uses. Thus, we need to choose $2^{M \cdot R}$ codewords in M dimensions, as in

$$\mathbf{x}_i = [x_{1,i}, x_{2,i}, \ldots, x_{M,i}],\ 1 \le i \le 2^{M \cdot R} \tag{67}$$

in order to keep the rate constant.

[21]Most practical channel codes are not block codes. But that a another story, and a very interesting one at that!

[22]Later in Section 4.3 a channel with multiple transmit and receive antennas is treated as a MIMO channel. Here, a MIMO channel is created by gathering together sequences of input and output uses.

At the receiver, a simple criterion for detection chooses the codeword $\mathbf{x}_i$ that minimizes $\|\mathbf{y} - \mathbf{x}_i\|^2$; that is, closest in Euclidean distance to the reception. The resulting index m of the minimizing $\mathbf{x}_m$ can be mapped back into the $M \cdot R$ bits. The noise immunity of a block code is accurately characterized by the minimum distance between pairs of codewords. Thus, the design of the codewords in (67) focuses on maximizing the minimum distance between pairs of codewords and also on *simultaneously* minimizing the transmit power.

The channel coding theorem states that if and only if $R \leq C$, there is at least one choice of codewords for each M such that, $P_e \to 0$ as $M \to \infty$. Observe that there is a *latency* of M channel uses since we must wait for a complete MIMO reception $\mathbf{y}$ before making a decision on the entire MR bits. This latency also grows to ∞ as $M \to \infty$.

Example (two-dimensional codewords). An alert reader might observe at this point that the signal constellation of Figure 1 is actually a degenerate channel code in $M = 2$ dimensions. That is true! It happens that in Figure 1 the two dimensions are the real and imaginary parts of a single *complex* channel use, but that is not mathematically significant because Euclidean geometry still applies. We can think of N complex-valued channel uses as equivalent to $2N$ real-valued channel uses. With this in mind, it is helpful to think of a block channel code as a kind of signal constellation in M dimensions. It is hard to conceptualize a higher-dimensionality space, and even harder to plot a higher-dimensionality signal constellation, so let's seek additional insight in two dimensions.

A signal constellation (degenerate channel code) in $M = 2$ dimensions is illustrated in Figure 11. In this and the subsequent figures, constellation points are represented by the "dots", in this case 196 of them implying that 7.61 bits of information are conveyed in two dimensions. The rate is thus $R = 7.61/2 = 3.8$ bits per channel use. Each constellation point is surrounded by a sphere (which reduces to a circle in two dimensions). Because the spheres touch one another, the minimum distance between code words (a measure of reliability) is twice the radius of the spheres.

In Figure 11 the code words have been laid out in a regular square grid, which is equivalent to concatenating two one-dimensional signal constellations, each with 14 points (note that $196 = 14^2$ and that $\log_2 14 = 3.8$). Without changing the peak transmitted power (represented by the large sphere surrounding the constellation), more points can be added to the constellation as in Figure 12. This allows us to increase the number of points in the constellation to 316, which can represent 8.3 bits. This is called *shaping* of the signal constellation. Observe that shaping can only be performed in two and higher dimensions, as the constellation is no longer the Cartesian product of two one-dimensional constellations as it was in Figure 11.

There is another trick that we can perform as illustrated in Figure 13. By simply rearranging the relative position of the spheres, again not affecting the peak power, the

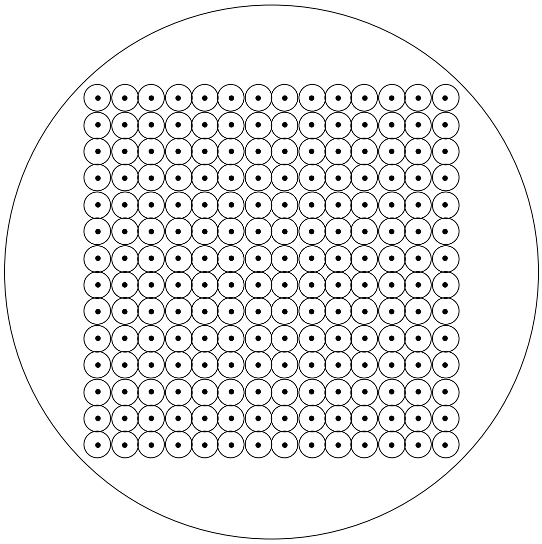

Figure 11: A two-dimensional square signal constellation with minimum distance unity and peak power constraint of $10^2 = 100$. The big circle represents the peak power constraint. The number of points in this constellation is 196.

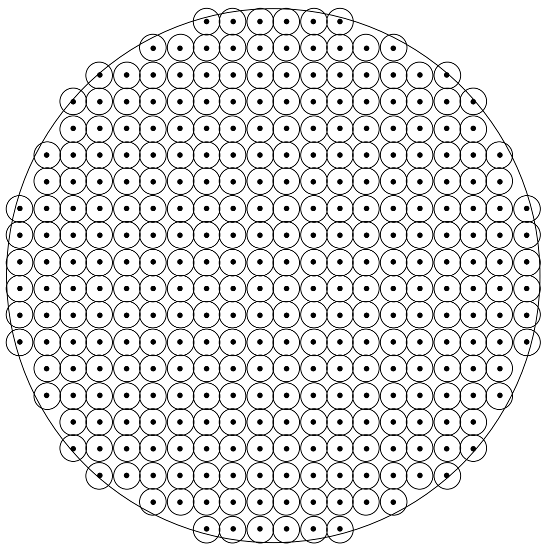

Figure 12: The two-dimensional constellation of Figure 11 with the addition of shaping. The number of points in this constellation is 316.

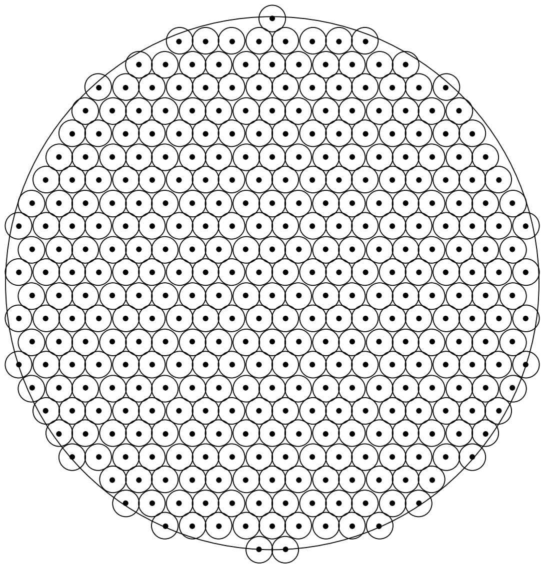

Figure 13: A two-dimensional constellation Figure 12 with the addition of coding, in which the spheres are rearranged to fit more within the power constraint sphere without changing the minimum distance. The number of points in this constellation is 362, communicating $\log_2 362 = 8.49$ bits per two dimensions.

number of codewords is increased further to 362, representing 8.49 bits. This simple trick captures the essence of channel coding. The fact is that *the spheres can be packed more tightly in two dimensions than in one dimension.*

The question at hand is, if $M = 2$ dimensions in a signal constellation (or equivalently block channel code) are better than $M = 1$, then are $M > 2$ dimensions better than $M = 2$. Let's not be timid in answering this question, and jump straight to $M = \infty$!

Example (sphere packing in higher dimensions). We can fairly easily arrive at an estimate of the number of signal constellation points that can be fit within a sphere with high dimensionality. The key observation is that the volume of a sphere in M dimensions is of the form $A_M \cdot r^M$, where r is the radius of the sphere. Knowing the radius (and hence volume) of a sphere that bounds the peak power, and also knowing the volume of a sphere surrounding each point in the signal constellation, we can obtain a bound on the number of points (spheres). Because the answer gives us the capacity formula of (40), this is both a plausibility argument for the channel coding theorem and a readily visualized interpretation.

The first step is to figure out the required radius for a sphere surrounding each point in the constellation. According to (65), there is a noise vector $\mathbf{N}$ added to each codeword $\mathbf{x}$ in the channel. Assuming that the components of $\mathbf{N}$ are independent, zero mean, and Gaussian with variance σ_N^2,

$$\begin{aligned} \|\mathbf{N}\|^2 &= \sum_{i=1}^{M} N_i^2 \\ E_N\left[\|\mathbf{N}\|^2\right] &= M \cdot \sigma_N^2 \,. \end{aligned} \tag{68}$$

According to our old friend the law of large numbers, as $M \to \infty$ high-probability noise vectors have a norm-squared that satisfies

$$\|\mathbf{N}\|^2 \approx M \cdot \sigma_N^2 \,. \tag{69}$$

Hence if we choose the radius of the sphere surrounding each signal constellation point to be slightly larger than $\sqrt{M \cdot \sigma_N^2}$ as $M \to \infty$, the probability of the noise $\mathbf{N}$ taking the reception out of the sphere approaches zero, and the error probability goes to zero. Conversely, if we choose the radius to be slightly smaller than $\sqrt{M \cdot \sigma_N^2}$, $\mathbf{N}$ takes us out of the sphere with increasing probability, and the error probability approaches unity. A lower bound on a radius that insures asymptotic reliability $\sqrt{M \cdot \sigma_N^2}$.

This argument explains why there is a sharp threshold (the channel capacity) between reliable and unreliable operation at the $M \to \infty$ asymptote. Now let's go further to explain the impact of a power constraint. If the peak *transmit* power is σ_X^2,

$$\|\mathbf{x}\|^2 \leq M \cdot \sigma_X^2 \tag{70}$$

then the peak *received* power (taking into account the noise) is asymptotically $M \cdot \left(\sigma_X^2 + \sigma_N^2\right)$. Thus a sphere that bounds all the small spheres surrounding signal constellation points must have radius $\sqrt{M \cdot \left(\sigma_X^2 + \sigma_N^2\right)}$.

The name of the game is now to pack the maximum number of small spheres into the large sphere without them overlapping. This is known as the *sphere packing* problem. The best we could possibly do is fill the entire space, in which case the number of small spheres (constellation points) is the volume of the big sphere divided by the volume of the small sphere,

$$
\begin{aligned}
2^{M\cdot R} &\le \frac{A_M \cdot \left(M\left(\sigma_X^2+\sigma_N^2\right)\right)^{M/2}}{A_M \cdot \left(M \cdot \sigma_N^2\right)^{M/2}} \\
&= \left(1+\frac{\sigma_X^2}{\sigma_N^2}\right)^{M/2}
\end{aligned} \tag{71}
$$

Thus, we arrive at the bound

$$
R \le \frac{1}{2}\log_2\left(1+\frac{\sigma_X^2}{\sigma_N^2}\right) , \tag{72}
$$

which is our familiar capacity formula. The preceding argument, while lacking rigor, suggests that the error probability goes to zero as $M \to \infty$ when (72) is satisfied, and goes to unity when it is not. A key point to dwell on is that the entire argument applies asymptotically as $M \to \infty$, and is certainly not strictly valid for any finite M.

These shaping and coding gains are relatively modest for $M = 2$ but increase as M increases. Approaching capacity has proven to be challenging in practice because of the exponential growth in receiver complexity as M increases. The only hope is to use a block code that is highly structured, where the coding and decoding are algorithmic in nature. A series of conceptual advances over the years have gradually chewed up the SNR gap between PAM and channel capacity on the Gaussian noise channel, culminating in *turbo codes* [10], which can often achieve near optimal performance in a practical way using current technology.

4.3 MIMO channel models

The additive Gaussian noise channel of the previous examples is accurate in a few situations, like deep-space communication with spacecraft, but generally too simplistic to be of value in most circumstances. This is particularly true of terrestrial wireless signals, which are subject to multipath distortion and interference effects. Since these radio systems are so important, it is useful to give a sense of how the previous results can be extended.

For terrestrial wireless, it is increasingly common to use more than one transmit and/or one receive antenna because this can significantly improve the reliability of communication in the presence of multipath fading. A generalization to (21) models the complex baseband channel as *multiple-input multiple-output*,

$$
\mathbf{Y}(t) = \sum \mathbf{G}(t-kT)\cdot \mathbf{A}_k + \mathbf{N}(t) \tag{73}
$$

In this model, all quantities are complex-valued (demodulation to baseband is embedded in the channel model), $\mathbf{A}_k$ is an n_x-dimensional input drawn from a constellation that is vector-valued (where $[\mathbf{A}_k]_i$ is a scalar complex-valued data symbol), $\mathbf{Y}(t)$ is an n_y-dimensional output vector signal (where $n_x \neq n_y$ in general), and $\mathbf{G}(t)$ is a $n_y \times n_x$ matrix-valued pulse. The physical significance is that the sequence $[\mathbf{A}_k]_i$ is the sequence of data symbols transmitted on the i-th channel (typically the i-th antenna), and $\mathbf{G}(t)$ captures not only the transmit pulse on each transmit channel, but also the response of the radio propagation channel (including multipath) from the i-th transmiter to the j-th receiver (typically j-th receive antenna).

After sampling in the receiver, and assuming that the channel is memoryless and fixed with time, (73) simplifies to

$$\mathbf{Y}_k = \mathbf{G} \cdot \mathbf{A}_k + \mathbf{N}_k \, . \tag{74}$$

Usually we can assume that the vector N_k is Gaussian with identically distributed and uncorrelated components, or $E[\mathbf{N}_k \mathbf{N}_k^\dagger] = \sigma_n^2 \cdot \mathbf{I}_{n_y}$ where "†" is the conjugate transpose operator and $\mathbf{I}_n$ is the $n \times n$ identify matrix. Further it may be reasonable to assume that the noise is white, or $E[\mathbf{N}_k \mathbf{N}_m^\dagger] = \mathbf{0}$ for $m \neq k$.

Rayleigh fading

The additive vector-noise model of (74) is a reasonable model for a wireless channel with an array of n_x transmit antennas and n_y receive antennas if $\mathbf{G}$ is a random matrix, typically one that is varying with time but slowly enough relative to sample k that it can be considered to be fixed-but-random over an appropriate periods of time. In a rich multipath environment where the line-of-sight (LOS) signal is insignificant but there are many signals arriving after reflections off various obstacles (ground, building, etc.), the Rayleigh fading model is accurate. In this model, the elements of the $\mathbf{G}$ matrix are independent and *not* identically distributed, each of the form

$$g_{i,j} = \alpha + i \cdot \beta \tag{75}$$

where α and β are independent and Gaussian distributed with variance $\sigma_{i,j}^2$. In this case, the magnitude $f_{i,j} = |g_{i,j}|$, which is the amplitude response of the channel from the i-th input to the j-th output, has a Rayleigh distribution (derived in Appendix C),

$$p_{F_{i,j}}(f_{i,j}) = \frac{1}{\sigma_{i,j}^2} \exp\left[-\frac{f_{i,j}^2}{2\sigma_{i,j}^2}\right] \, , \; f \geq 0 \, . \tag{76}$$

For this reason, this channel model is called *Rayleigh fading.*

Fixed and known channel receiver design

Receiver design depends on what assumptions about the channel are reasonable. The memoryless assumption of (74) presumes that each channel output vector depends on only one channel input data symbol vector. The time-invariant assumption presumes that the channel matrix does not change with time, at least over a period of time of interest. Usually it is assumed that the channel matrix **G** is known at the receiver, because it can be accurately estimated there[23]. It may also be reasonable to assume that **G** is known at the transmitter as well if the receiver observes the channel outputs, estimates **G** accurately, and then transmits that information back to the transmitter, all quickly enough that **G** has not had time to change by much.

To illustrate a receiver design for PAM, assume that our approach is to estimate $\hat{\mathbf{A}}$ of data symbol **A** of the form

$$\hat{\mathbf{A}} = \mathbf{C} \cdot \mathbf{Y} \tag{77}$$

In some sense, the goal of **C** is to invert the channel matrix **G**, but of course that direct objective may be impossible if **G** is not invertible[24]. The question is, what **C** should we use? An simple criterion is to minimize the error

$$\begin{aligned} \mathbf{\Delta} &= \hat{\mathbf{A}} - \mathbf{A} \\ &= \mathbf{C} \cdot \mathbf{Y} - \mathbf{A} \end{aligned} \tag{78}$$

in some sense. It is tempting to minimize $\|\mathbf{\Delta}\|^2$, but this only gives us n_y equations in $n_x n_y$ variables (the elements of $n_x \times n_y$ matrix **C**). Therefore, let's attack the more ambitious goal of minimizing the $n_x \times n_x$ autocorrelation matrix of the error,

$$\mathbf{R}_e = E[\mathbf{\Delta}\mathbf{\Delta}^\dagger] \tag{79}$$

The minimization of (79) is considered in Appendix D. It is shown that the optimum matrix $\mathbf{C}_{\text{opt}}$ is given by (129),

$$\mathbf{C}_{\text{opt}} = (\mathbf{G}^\dagger \mathbf{G} + \sigma_n^2 \cdot \mathbf{I}_{n_x})^{-1} \mathbf{G}^\dagger \tag{80}$$

and the resulting minimum error autocorrelation matrix is given by (130),

[23]This depends on adaptive estimation techniques not discussed here. It also depends on the assumption that **G** varies slowly enough that it can be accurately estimated before it has a chance to change significantly.

[24]For example, **G** may not be symmetric (the number of transmit and receive antennas is different), or even if it is symmetric it may be singular.

$$\mathbf{R}_{e,\min} = \sigma_n^2 \cdot (\mathbf{G}^\dagger \mathbf{G} + \sigma_n^2 \cdot \mathbf{I}_{n_x})^{-1} \,. \tag{81}$$

We can gain an additional understanding of this solution by the following examples.

Example (low SNR). As the SNR gets low because $\sigma_n^2 \to \infty$,

$$\mathbf{C}_{\text{opt}} \to \frac{1}{\sigma_n^2} \cdot \mathbf{G}^\dagger \tag{82}$$

$$\mathbf{R}_{e,\min} \to \mathbf{I}_{n_x} \,. \tag{83}$$

The estimator is proportional to $\mathbf{G}^\dagger$, which is a *matrix matched filter* and the resulting error autocorrelation approaches diagonal, implying that the different components of error become uncorrelated (because they are dominated by the noise, which is assumed to have a similar property).

Example (high SNR). As the SNR becomes high because $\sigma_n^2 \to 0$, assuming that $\mathbf{G}^\dagger\mathbf{G}$ is invertible[25],

$$\mathbf{C}_{\text{opt}} \to (\mathbf{G}^\dagger \mathbf{G})^{-1} \mathbf{G}^\dagger \tag{84}$$

$$\mathbf{R}_{e,\min} \to \sigma_n^2 \cdot (\mathbf{G}^\dagger \mathbf{G})^{-1} \,. \tag{85}$$

$$\hat{\mathbf{A}} = \mathbf{C}_{\text{opt}} \mathbf{G} \mathbf{Y} \to \mathbf{Y} \,. \tag{86}$$

The estimator still begins with a matrix matched filter, but then applies a transformation that eliminates the interference between the different channels (different antennas), so that the estimate approaches $\hat{\mathbf{A}}$ as expected. Note also that the error becomes small, but small $\mathbf{G}$ (large propagation attenuation) tends to increase the error for a given noise level.

Example (optimal two-antenna diversity combining). It is helpful to consider a couple of low-dimensionality examples. Consider the case of a single transmit antenna ($n_x = 1$) and two receive antennas ($n_y = 2$). Then $\mathbf{G}$ is of the form

[25] This requires two things: first $n_y \geq n_x$ (the number of receive antennas is greater than the number of transmit antennas), and secondly $\mathbf{G}$ is a full rank matrix. When $\mathbf{G}^\dagger\mathbf{G}$ is not invertible, the solution becomes a *pseudo-inverse*.

$$
\begin{aligned}
\mathbf{G} &= \begin{bmatrix} g_{1,1} \\ g_{2,1} \end{bmatrix} \\
\mathbf{G}\mathbf{G}^\dagger &= \begin{bmatrix} |g_{1,1}|^2 & g_{1,1}g_{2,1}^* \\ g_{1,1}^* g_{2,1} & |g_{2,1}|^2 \end{bmatrix} \\
\mathbf{G}^\dagger\mathbf{G} &= \left[|g_{1,1}|^2 + |g_{2,1}|^2\right]
\end{aligned}
\tag{87}
$$

$$
\begin{aligned}
\mathbf{C}_{\text{opt}} &= \frac{\mathbf{G}^\dagger}{\mathbf{G}^\dagger\mathbf{G} + \sigma_n^2} = \frac{\begin{bmatrix} g_{1,1}^* & g_{2,1}^* \end{bmatrix}}{|g_{1,1}|^2 + |g_{2,1}|^2 + \sigma_n^2} \\
\mathbf{R}_{e,\min} &= \frac{\sigma_n^2}{\mathbf{G}^\dagger\mathbf{G} + \sigma_n^2} = \frac{\sigma_n^2}{|g_{1,1}|^2 + |g_{2,1}|^2 + \sigma_n^2}\,.
\end{aligned}
\tag{88}
$$

The receiver processing forms an estimate of single data symbol $\hat{A}$ by calculating

$$
\hat{A} = \frac{g_{1,1}^* y_1 + g_{2,1}^* y_2}{|g_{1,1}|^2 + |g_{2,1}|^2 + \sigma_n^2}\,.
\tag{89}
$$

Processing (89) is called *optimal diversity combining.* The idea is to combine the two antenna signals in such a way that the reliability is improved due to the independent fading from the single transmitter antenna to the two receiving antennas. The merit of the technique can be seen in $\mathbf{R}_{e,\min}$, which depends on the channel $\mathbf{G}$ only through the value of $|g_{1,1}|^2 + |g_{2,1}|^2$. The estimation error gets large only when the magnitudes of *both* signals from the two receive antennas is small. The receiver processing takes full advantage of the signal power received by both antennas.

Example (two transmit antennas). Consider the case of two transmit antennas ($n_x = 2$) and a single receive antenna ($n_y = 1$). Then $\mathbf{G}$ is of the form

$$
\begin{aligned}
\mathbf{G} &= \begin{bmatrix} g_{1,1} & g_{1,2} \end{bmatrix} \\
\mathbf{G}\mathbf{G}^\dagger &= \left[|g_{1,1}|^2 + |g_{1,2}|^2\right] \\
\mathbf{G}^\dagger\mathbf{G} &= \begin{bmatrix} |g_{1,1}|^2 & g_{1,1}^* g_{1,2} \\ g_{1,1} g_{1,2}^* & |g_{1,2}|^2 \end{bmatrix}
\end{aligned}
\tag{90}
$$

$$
\begin{aligned}
\mathbf{C}_{\text{opt}} &= \frac{1}{\sigma_n^2 + |g_{1,1}|^2 + |g_{1,2}|^2} \cdot \begin{bmatrix} g_{1,1}^* \\ g_{1,2}^* \end{bmatrix} \\
\mathbf{R}_{e,\min} &= \frac{1}{\sigma_n^2 + |g_{1,1}|^2 + |g_{1,2}|^2} \cdot \begin{bmatrix} \sigma_n^2 + |g_{1,2}|^2 & -g_{1,1}^* g_{1,2} \\ -g_{1,1} g_{1,2}^* & \sigma_n^2 + |g_{1,1}|^2 \end{bmatrix}\,.
\end{aligned}
\tag{91}
$$

The interesting aspect of this example is that the receiver processing turns the single antenna signal into a two-component signal to correspond to the two data symbols A_1 and A_2 using a matrix matched filter. Not surprisingly, the spatial diversity of the last example is missing. Examining the diagonal elements of $\mathbf{R}_{e,\min}$, the error in A_1 being small depends on $|g_{1,1}|^2 >> |g_{1,2}|^2$, and the error in A_2 being small depends on $|g_{1,2}|^2 >> |g_{1,1}|^2$. Obviously both of these conditions cannot be satisfied simultaneously. Thus, the receiver may do a good job of estimating A_1 or A_2 individually, but not both at the same time.

This suggests that using two transmit antennas with a single receive antenna is not advantageous. Is there a way to achieve the results of diversity combining with two transmit antennas rather than two receive antennas? Yes! Later, we will illustrate a way to replicate the $(n_x = 1, n_y = 2)$ optimal diversity combining using the $(n_x = 2, n_y = 1)$ physical channel combined with Alamouti space-time coding in the transmitter. We can do this in a way that the crosstalk between the two channels is totally eliminated, without the transmitter even having to know $\mathbf{G}$.

Channel capacity

The vector PAM channel model of (74) presumes a particular modulation scheme. What is the fundamental channel capacity limit? This can be explored by generalizing (74) to

$$\mathbf{Y}_k = \mathbf{G} \cdot \mathbf{X}_k + \mathbf{N}_k \,. \tag{92}$$

For this purpose, assume that $\mathbf{G}$ is known at the transmitter and receiver if the receiver observes the channel outputs, estimates $\mathbf{G}$ accurately, and then transmits that information back to the transmitter, all quickly enough that $\mathbf{G}$ has not had time to change by much. In that case, capacity as well as channel coding design can assume that $\mathbf{G}$ is fixed and known, but drawn from a random ensemble. The technique is then to calculate capacity conditional on a fixed-and-known $\mathbf{G}$ and then average that capacity over the random ensemble of $\mathbf{G}$. If $\mathbf{G}$ is not known accurately, or not at all, then more complicated procedures have to be followed.

Let's illustrate this by displaying the capacity of the MIMO channel for fixed-and-known $\mathbf{G}$ [11]. Assume that both $\mathbf{X}_k$ and $\mathbf{N}_k$ are independent and identically distributed over k and have autocorrelation matricies

$$E[\mathbf{X}_k \mathbf{X}_k^\dagger] = \mathbf{R}_x \text{ and } E[\mathbf{N}_k \mathbf{N}_k^\dagger] = \mathbf{R}_n \,. \tag{93}$$

If we wish to impose a power limit on the input, the most reasonable approach is to allow the power of the components of $\mathbf{X}_k$ to be different, but to be limited in variance overall,

$$\mathrm{tr}\{\mathbf{R}_x\} \leq P_T \tag{94}$$

where "tr" denotes the matrix trace (sum of diagonal elements). This allows the transmitter to allocate more power to any sub-channels for which propagation conditions are more favorable.

The method for calculating channel capacity for a MIMO channel is essentially the same as earlier. The joint probability $p_{\mathbf{X},\mathbf{Y}}(\mathbf{x}, \mathbf{y})$ is a scalar-valued function, so the definitions of average uncertainty, equivocation, transinformation, etc. are unchanged. The underlying ensemble random variables just happen to be vector-valued.

It is straightforward to show that the mutual information per channel use is

$$I(\mathbf{X}, \mathbf{Y}) = H(\mathbf{Y}) - H(\mathbf{N}) \tag{95}$$

Since $H(\mathbf{N})$ does not depend on the distribution of $\mathbf{X}$, the channel capacity can be determined by maximizing $H(\mathbf{Y})$ over all distributions of $\mathbf{X}$ that satisfy the power constraint. Calculating the correlation matrix of $\mathbf{Y}$,

$$E[\mathbf{Y}\mathbf{Y}^\dagger] = E[(\mathbf{G}\mathbf{X} + \mathbf{N})(\mathbf{G}\mathbf{X} + \mathbf{N})^\dagger] = \mathbf{G}\mathbf{R}_x\mathbf{G}^\dagger + \mathbf{R}_n \,. \tag{96}$$

A variance-constrained complex-valued Gaussian random vector $\mathbf{Y}$ has average uncertainty (Appendix E)

$$H(\mathbf{Y}) = \log_2 (\pi e)^n |\mathbf{R}_y| \,, \tag{97}$$

where $|\mathbf{K}|$ is the determinant of matrix $\mathbf{K}$. As in the scalar case, this is also the maximum average uncertainty for any variance-constrained complex-valued random vector. Therefore, we can bound the transinformation,

$$\begin{aligned} I(\mathbf{X}, \mathbf{Y}) &\leq \log_2 \left| \mathbf{G}\mathbf{R}_x\mathbf{G}^\dagger + \mathbf{R}_n \right| - \log_2 |\mathbf{R}_n| \\ &= \log_2 \left| \mathbf{G}\mathbf{R}_x\mathbf{G}^\dagger\mathbf{R}_n^{-1} + \mathbf{I}_{n_y} \right| \,. \end{aligned} \tag{98}$$

For example, if (as is often the case) it is reasonable to assume that the noise is uncorrelated across receive components, or $\mathbf{R}_n = \sigma_n^2 \mathbf{I}_{n_y}$, then $|\mathbf{R}_n| = \sigma_n^{2n_y}$ and

$$I(\mathbf{X},\mathbf{Y}) \leq \log_2 \left| \frac{1}{\sigma_n^2} \mathbf{G}\mathbf{R}_x\mathbf{G}^\dagger + \mathbf{I}_{n_y} \right| . \tag{99}$$

Since the upper bound (99) can in principle be maximized over $\mathbf{R}_x$ subject to the power constraint, the resulting upper bound is realized when the input $\mathbf{X}$ is Gaussian, so this determines the capacity.

The right side of (99) is easily evaluated for the special case where the input signal power is equally distributed across transmit components, or $\mathbf{R}_x = \frac{P_T}{n_x} \cdot \mathbf{I}_{n_x}$. Since this is a specific input distribution,, not necessarily the optimum, this yields a lower bound on capacity,

$$C \geq \log_2 \left| \frac{P_T}{n_x \sigma_n^2} \mathbf{G}\mathbf{G}^\dagger + \mathbf{I}_{n_y} \right| . \tag{100}$$

Of course, for any given $\mathbf{G}$ the maximum of (99) can be determined, yielding the exact capacity C. This tells us not only that the optimum input vector is Gaussian, but also the capacity-achieving covariance matrix, which will typically not allocate power equally across input components (the components which less attenuation through the channel will be weighted more heavily) and will also have correlated components to take advantage of the coupling intrinsic in channel matrix $\mathbf{G}$.

Capacity with Rayleigh fading

Thus far the capacity calculation has assumed a fixed and known channel matrix $\mathbf{G}$. What is the effect of Rayleigh fading? It is easy to estimate the lower bound on capacity of (100) when averaged over a random ensemble of G. As a consequence of the law of large numbers,

$$\frac{1}{n_x} \cdot \mathbf{G}\mathbf{G}^\dagger \rightarrow \mathbf{I}_{n_y} \tag{101}$$

as n_x gets large.

Substituting (101) in (100),

$$\begin{aligned} C &\geq \log_2 \left| \left(\frac{P_T}{\sigma_n^2} + 1 \right) \mathbf{I}_{n_y} \right| \\ &= \log_2 \left(\frac{P_T}{\sigma_n^2} + 1 \right)^{n_y} \\ &= n_y \cdot \log_2 \left(1 + \frac{P_T}{\sigma_n^2} \right) \end{aligned} \tag{102}$$

This is the familiar $\log_2(1+\text{SNR})$ form, but it tells us that for a fixed number of channel outputs n_y the capacity is at least proportional to n_y as $n_x \to \infty$. For a Rayleigh fading channel, as the number of transmit antennas is increased the capacity is asymptotically at least proportional to the number of receive antennas.

As another example, when the number of transmit and receive antennas is equal ($n_x = n_y = n$), the capacity increases at least as fast as n [12]. This implies that it is possible to achieve dramatic increases in bit rates and spectral efficiency with the use of antenna arrays. It also tells us that fundamentally multipath distortion is actually a desirable phenomenon because it improves the overall reliability of propagation.

Coding for the MIMO channel

Historically, for many years multipath distortion was treated as an impairment to be overcome. The large MIMO capacity of such a channel, however, is substantially larger than a channel without multipath, as long as there are multiple input and output antennas and as long as the antenna array geometry results in uncorrelated channel complex-valued gains. Multipath does not have to be an impairment, but it can actually enhance spectral efficiency. The intuition behind this is that multiple input and output antennas (if they are spaced at least a half-wavelength apart) give us uncorrelated looks at the channel, thus giving us an opportunity to communicate more data in parallel using these parallel paths. In practice, this is exploited by using channel coding that generalizes the scalar channel, in the sense that it defines code words that are a vector of input samples in time, but also in space. This is called *space-time* coding [13], and in the past decade it has resulted in dramatic increases in spectral efficiency for multipath channels [12].

Example (Alamouti code). Space-time-coding can be illustrated in a simple way by the Alamouti code [14], which is incorporated into the IEEE 802.11a WiFi standard. This coding assumes $n_x = 2$ (two transmit antennas) and $n_y = 1$ (one receive antenna).

$$\begin{bmatrix} y_k \end{bmatrix} = \begin{bmatrix} g_1 & g_2 \end{bmatrix} \begin{bmatrix} x_{1,k} \\ x_{2,k} \end{bmatrix} \tag{103}$$

The Alomouti code groups two successive channel uses, say $k = 1$ and $k = 2$, to communicate two data symbols A_1 and A_2,

$$\begin{aligned} \begin{bmatrix} y_1 \end{bmatrix} &= \begin{bmatrix} g_1 & g_2 \end{bmatrix} \begin{bmatrix} A_1 \\ A_2 \end{bmatrix} \\ \begin{bmatrix} y_2 \end{bmatrix} &= \begin{bmatrix} g_1 & g_2 \end{bmatrix} \begin{bmatrix} -A_2^* \\ A_1^* \end{bmatrix} \end{aligned} \tag{104}$$

This is a simple form of space-time code because each of the two data symbols is repeated redundantly in both space (the two antennas) and in time (two successive symbol intervals). Writing the two output samples in a single vector, and conjugating the second for the convenience of later processing,

$$\begin{bmatrix} y_1 \\ y_2^* \end{bmatrix} = \begin{bmatrix} g_1 & g_2 \\ g_2^* & -g_1^* \end{bmatrix} \begin{bmatrix} A_1 \\ A_2 \end{bmatrix} \tag{105}$$

This is the model of an effective ($n_x = 2$, $n_y = 2$) channel which has one channel use for each pair of channel uses of the original channel. The key simplification is that the $\mathbf{G}$ matrix of this effective channel is unitary,

$$\mathbf{G}^\dagger \mathbf{G} = \mathbf{G}\mathbf{G}^\dagger = (|g_1|^2 + |g_2|^2) \cdot \mathbf{I}_2 \,. \tag{106}$$

Thus, a matrix matched filter in the receiver diagonalizes the channel, or eliminates any crosstalk from one transmit signal into the other as seen by the receiver. The optimal receiver matrix and resulting error correlation are given by

$$\begin{aligned} \mathbf{C}_{\text{opt}} &= \frac{1}{\sigma_n^2 + |g_1|^2 + |g_2|^2} \cdot \begin{bmatrix} g_1^* & g_2 \\ g_2^* & g_1 \end{bmatrix} \\ \mathbf{R}_{e,\min} &= \frac{\sigma_n^2}{\sigma_n^2 + |g_1|^2 + |g_2|^2} \cdot \mathbf{I}_2 \,. \end{aligned} \tag{107}$$

Comparing the error correlation matrix $\mathbf{R}_{e,\min}$ of (107) with the receive diversity of (88), we see that they are identical. Thus, the estimation error performance of the Alamouti code is identical to two-receive-antenna optimal diversity combining. The difference is that two receive antennas are replaced by two transmit antennas, and a portion of the processing is performed by a space-time code in the transmitter.

Alamouti coding does pay a penalty in transmit power relative to optimal diversity combining. The total transmit power in the two antennas is double, which increases the interference into any other wireless communication systems sharing the same frequency, time, and space coordinates.

---

5 Conclusions

With the exception of some results in Section 4.3, all of the principles we have outlined arose from work performed in the 1940s. They established a theoretical basis for digital communication, which first made their appearance in commercial systems in the 1960's and have since has become practically universal in new communication system designs. Throughout the 1950's, 1960's, 1970's most of the research focused on improving digital communication, including dealing with dispersive and time-varying channels and achieving rates closer to channel capacity. Advances in microelectronics were crucial for realizing many of the computationally intensive algorithms that resulted.

In parallel, in the 1970's fiber optic communication arose. Over most of fiber's history most progress has been based on improvements to physical devices and media rather than communication theories such as have been espoused here. Fiber has come to dominate long-distance communication, including under-sea as well as terrestrial. Meantime cable-based electronic communication has remained important on existing media for the so-called "last kilometer", for wireless local access to the fiber backbone, and for mobility. Thus, the importance of radio communications has increased even as fiber optics has gained dominance in the core of the network. Most research since the 1990's has focused on wireless multiple access, multipath and fading, interference, and mobility.

Acknowledgement

The author is grateful to Gerry Harp of the SETI Institute, who provided very useful comments on an earlier draft of this report, particularly in terms of making it more comprehensible to physicists.

Appendix A: Notation review

A random variable X can be either discrete, taking on values from $\{x_k,\ 1 \le k \le K\}$, or continuous, taking on values drawn from $(-\infty, \infty)$. In some cases complex-valued random variables such as $Z = U + i \cdot V$ will be encountered, where U and V are real-valued random variables. The complex conjugate is $Z^* = U - i \cdot V$.

In the discrete case, X has a *probability mass function* $p_X(x)$ with the definition

$$p_X(x_k) \equiv \Pr\{X = x_k\} \tag{108}$$

and in the continuous case X has a *probability density function* $p_X(x)$ with the property that

$$\Pr\{a \le X < b\} = \int_a^b p_X(x)\,\mathrm{d}x\,. \tag{109}$$

A function $f(\cdot)$ defines a new random variable $Y = f(X)$. The expected value of Y is the average over the X ensemble,

$$\begin{aligned} E_X(Y) &= \sum_{k=1}^{K} f(x_k) \cdot p_X(x_k) \\ E_X(Y) &= \int f(x) \cdot p_X(x)\,\mathrm{d}x\,. \end{aligned} \tag{110}$$

In particular the mean and variance are defined as

$$\begin{aligned} \mu_X &= E_X(X) \\ \sigma_X^2 &= E_X|X - \mu_X|^2\,. \end{aligned} \tag{111}$$

Two random variables X and Y have a *joint* probability mass or density function $p_{X,Y}(x, y)$ and a conditional mass or density function

$$p_{Y|X}(y|x) = \frac{p_{X,Y}(x, y)}{p_X(x)} \tag{112}$$

where (112) has the interpretation as the mass or density function of Y with prior knowledge of $X = x$.

A *random process* is a sequence of random variables X_k in the discrete-time case and and set of random variables indexed by a real number t, denoted by $X(t)$, in the continuous-time case. The autocorrelation of a complex-valued wide-sense stationary random process is

$$\begin{aligned} R_X(m) &= E_X\left[X_{k+m} \cdot X_k^*\right] \\ R_X(\tau) &= E_X\left[X(t+\tau) \cdot X^*(t)\right] \end{aligned} \tag{113}$$

assumed to not be a function of k or t.

The *power spectrum* of a wide-sense stationary random process is the Fourier transform of the autocorrelation,

$$\begin{aligned} S_X(f) &= \sum_m R_X(m) \cdot e^{-i2\pi f m} \\ S_X(f) &= \int R_X(\tau) \cdot e^{-i2\pi f \tau} \, \mathrm{d}\tau \, . \end{aligned} \tag{114}$$

Continuous-time *white noise* is particularly important and easy to deal with, where

$$\begin{aligned} R_X(\tau) &= N_0 \cdot \delta(\tau) \\ S_X(f) &= N_0 \, . \end{aligned} \tag{115}$$

Appendix B: Maximum average uncertainty

The inequality

$$\log_2 y \leq 1 - y \text{ for } y > 0 \tag{116}$$

with equality if and only if $y = 1$ is useful in deriving some bounds on average uncertainty.

What discrete random variable X has the largest average uncertainty $H(x)$? Let $p_X(x)$ be a probability mass function for a discrete random variable X with K distinct values, and let $g(x)$ be any other probability mass function with the same values of x. Then

$$
\begin{aligned}
\sum_x p_X(x) \cdot \log_2 \frac{g(x)}{p_X(x)} &\le \sum_x p_X(x) \cdot \left(1 - \frac{g(x)}{p_X(x)}\right) \\
&= 1 - 1 = 0
\end{aligned}
\tag{117}
$$

with equality if and only if $p_X(s) \equiv g(x)$. Then

$$
\sum_x p_X(x) \cdot \log_2 \frac{1}{p_X(x)} \le \sum_x p_X(x) \cdot \log_2 \frac{1}{g(x)} \tag{118}
$$

and substituting $g(x) = 1/K$ where K is the number of distinct values of X, the inequality becomes

$$
H(X) \le \log_2 K \tag{119}
$$

Equality occurs in (119) when the K values of X are equally likely.

What continuous zero-mean random variable X with variance σ_x^2 has the largest $H(X)$? Let X be a zero mean random variable with probability density function $p_X(x)$ and variance σ_x^2, and let $g(x)$ be any other zero mean probability density function. For simplicity assume that $p_X(x) \neq 0$ and $g(x) \neq 0$ for $-\infty < x < \infty$. Then

$$
\begin{aligned}
\int p_X(x) \log_2 \frac{g(x)}{p_X(x)} \, \mathrm{d}x &\le \int p_X(x) \cdot \left(1 - \frac{g(x)}{p_X(x)}\right) \mathrm{d}x \\
&= 1 - 1 = 0
\end{aligned}
\tag{120}
$$

with equality if and only if $p_X(x) \equiv g(x)$. Hence

$$
H(X) \le \int p_X(x) \log_2 \frac{1}{g(x)} \, \mathrm{d}x \, . \tag{121}
$$

Substituting a zero-mean Gaussian distribution with variance σ_x^2 for $g(x)$, this becomes

$$
\begin{aligned}
H(X) &\le \int p_X(x) \cdot \left(\log_2 \sigma_X \sqrt{2\pi} + \frac{x^2}{2\sigma_x^2} \cdot \log_2 e\right) \mathrm{d}x \\
&= \log_2 \sigma_x \sqrt{2\pi e} \, .
\end{aligned}
\tag{122}
$$

Equality holds in (122) if and only if $p_X(x)$ is Gaussian.

Appendix C: Rayleigh distribution

Let U and V be independent identically distributed zero-mean Gaussian random variables with variance σ^2, and let the random variable $R = \sqrt{U^2 + V^2}$ be the magnitude of complex-valued random varaible $U + i \cdot V$. What is the probability density of R?

The distribution function of R is

$$\begin{aligned}\Pr\{R \le r\} &= \Pr\{\sqrt{U^2+V^2} \le r\} \\ &= \int_{-r}^{r} \mathrm{d}u \int_{-\sqrt{r^2-u^2}}^{\sqrt{r^2-u^2}} \mathrm{d}v \, \frac{1}{2\pi\sigma^2} \cdot e^{-(u^2+v^2)/2\sigma^2} \\ &= \int_{-r}^{r} \mathrm{d}v \frac{1}{\sigma\sqrt{2\pi}} \cdot e^{-u^2/2\sigma^2} \cdot \mathrm{Erf}\left(\frac{\sqrt{r^2-u^2}}{\sigma\sqrt{2}}\right) .\end{aligned} \tag{123}$$

The density of R is obtained by differentiating this distribution function,

$$p_R(r) = \frac{\mathrm{d}}{\mathrm{d}r}\Pr\{R \le r\} = \frac{r \cdot e^{-r^2/2\sigma^2}}{\sigma^2} . \tag{124}$$

Appendix D: Matrix optimization

Our goal is to choose $\mathbf{C}$ so as to minimize the autocorrelation matrix of (79). Expanding $\mathbf{R}_e$,

$$\begin{aligned}\mathbf{R}_e &= E[\mathbf{E}\mathbf{E}^\dagger] \\ &= \mathbf{C}\mathbf{R}_y\mathbf{C}^\dagger + \mathbf{I}_{n_x} - \mathbf{G}^\dagger\mathbf{C}^\dagger - \mathbf{C}\mathbf{G} \\ &= (\mathbf{C} - \mathbf{G}^\dagger\mathbf{R}_y^{-1})\,\mathbf{R}_y\,(\mathbf{C} - \mathbf{G}^\dagger\mathbf{R}_y^{-1})^\dagger + \mathbf{I}_{n_x} - \mathbf{G}^\dagger\mathbf{R}_y^{-1}\mathbf{G} .\end{aligned} \tag{125}$$

In (125) we have made several definitions and assumptions. We assume $\mathbf{G}$ is known, the noise vector has independent identically distributed components,

$$E[\mathbf{N}\mathbf{N}^\dagger] = \sigma_n^2 \cdot \mathbf{I}_{n_y} \tag{126}$$

and the input vector has independent identically distributed components with unit variance,

$$E[\mathbf{A}\mathbf{A}^\dagger] = \mathbf{I}_{n_x} . \tag{127}$$

Further, the autocorrelation matrix of the channel output is defined as

$$\begin{aligned}\mathbf{R}_y &= E[\mathbf{Y}\mathbf{Y}^\dagger] \\ &= \mathbf{G}\mathbf{G}^\dagger + \sigma_n^2\,\mathbf{I}_{n_y}\,.\end{aligned} \tag{128}$$

The first term in (125) is eliminated, which minimizes every element of $\mathbf{R}_e$, by choosing

$$\begin{aligned}\mathbf{C}_{\text{opt}} &= \mathbf{G}^\dagger\mathbf{R}_y^{-1} \\ &= \mathbf{G}^\dagger(\mathbf{G}\mathbf{G}^\dagger + \sigma_n^2\,\mathbf{I}_{n_y})^{-1} \\ &= (\mathbf{G}^\dagger\mathbf{G} + \sigma_n^2\mathbf{I}_{n_x})^{-1}\mathbf{G}^\dagger\end{aligned} \tag{129}$$

The last two terms in (129) are equivalent, as is easily shown by multiplying through both sides by the inverses. Finally, the minimum error correlation matrix is

$$\begin{aligned}\mathbf{R}_{e,\min} &= \mathbf{I}_{n_x} - \mathbf{G}^\dagger\mathbf{R}_y^{-1}\mathbf{G} \\ &= \mathbf{I}_{n_x} - \mathbf{C}_{\text{opt}}\mathbf{G} \\ &= \sigma_n^2\,(\mathbf{G}^\dagger\mathbf{G} + \sigma_n^2\mathbf{I}_{n_x})^{-1}\,.\end{aligned} \tag{130}$$

The last form in (130) follows directly from substituting for $\mathbf{C}_{\text{opt}}$ from the last form in (129).

Appendix E: Joint Gaussian average uncertainty

The average uncertainty of a multivariate Gaussian random variable $\mathbf{X}$ with dimension n, mean μ, and covariance matrix $\mathbf{\Sigma}$ is now determined. First assume that $\mathbf{X}$ is real-valued, in which case its uncertainty is

$$-\log_2 p_{\mathbf{X}}(\mathbf{x}) = \frac{1}{2}\log_2(2\pi)^n|\mathbf{\Sigma}| + \frac{1}{2}\cdot(\log_2 e)(\mathbf{X}-\mu)^T\mathbf{\Sigma}^{-1}(\mathbf{X}-\mu)\,. \tag{131}$$

It is readily shown that[26]

$$E_{\mathbf{X}}\,(\mathbf{X}-\mu)^T\mathbf{\Sigma}^{-1}(\mathbf{X}-\mu) = n \tag{132}$$

[26]For example, replace $\mathbf{\Sigma}$ with its spectral representation in terms of its positive-real-valued eigenvalues and an orthonormal set of eigenvectors.

so that

$$H(\mathbf{X}) = \frac{1}{2}\log_2(2\pi e)^n|\mathbf{\Sigma}|\,. \tag{133}$$

Now assume that $\mathbf{X}$ is complex-valued and zero-mean, and form a $2n$-dimensional real-valued vector $\tilde{\mathbf{X}}$ out of the real and imaginary parts,

$$\begin{aligned}\mathbf{X} &= \mathbf{X}_r + i\cdot\mathbf{X}_i \\ \tilde{\mathbf{X}} &= \begin{bmatrix}\mathbf{X}_r \\ \mathbf{X}_i\end{bmatrix}\,.\end{aligned} \tag{134}$$

In (134), for simplicity the real- and imaginary-parts can be assumed to be independent and identically distributed[27], each with covariance matrix $\mathbf{\Lambda}$,

$$\begin{aligned}&E_{\mathbf{X}}\left[\mathbf{X}_r\mathbf{X}_r^T\right] = E_{\mathbf{X}}\left[\mathbf{X}_i\mathbf{X}_i^T\right] = \mathbf{\Lambda} \\ &E_{\mathbf{X}}\left[\mathbf{X}_r\mathbf{X}_i^T\right] = \mathbf{0} \\ &\mathbf{R}_x = E_{\mathbf{X}}\left[\mathbf{X}\mathbf{X}^\dagger\right] = 2\mathbf{\Lambda} \\ &\mathbf{\Sigma} = \begin{bmatrix}\mathbf{\Lambda} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Lambda}\end{bmatrix} \\ &|\mathbf{\Sigma}| = |\mathbf{\Lambda}|^2\,.\end{aligned} \tag{135}$$

Invoking (133) with n replaced by $2n$, the average uncertainty is then

$$\begin{aligned}H(\mathbf{X}) &= \log_2(2\pi e)^n|\mathbf{\Lambda}| \\ &= \log_2(\pi e)^n|\mathbf{R}_x|\,.\end{aligned} \tag{136}$$

[27] This choice can also be shown to maximize the average uncertainty. Unlike a real-valued Gaussian vector, a complex-valued Gaussian vector's statistics are not completely characterized by its covariance, and these additional assumptions, which are termed *circular symmetry*, are needed.

References

[1] J. G. Proakis, *Digital Communication.* Osborne-McGraw-Hill, 2001,

[2] J. R. Barry, E. A. Lee and D. G. Messerschmitt, *Digital Communication*, 3rd ed. Boston: Kluwer Academic Publishers, 2004.

[3] T. L. Wilson, "The search for extraterrestrial intelligence," *Nature*, vol. 409, pp. 1110-1114, 2001.

[4] R. C. Dixon, *Spread Spectrum Systems.* John Wiley and Sons, Inc. New York, NY, USA, 1990.

[5] R. L. Peterson, R. E. Ziemer and D. E. Borth, *Introduction to Spread-Spectrum Communications.* Prentice Hall, Englewood Cliffs, NJ, 1995.

[6] L. B. Milstein, "Interference rejection techniques in spread spectrum communications," *Proceedings of the IEEE*, vol. 76, pp. 657-671, 1988.

[7] B. Oliver, J. Pierce and C. Shannon, "The Philosophy of PCM," *Proceedings of the IRE*, vol. 36, pp. 1324-1331, 1948.

[8] C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, pp. 379-423, 1948.

[9] T. Cover and J. Thomas, *Elements of Information Theory.* New York: John Wiley and Sons, 1991.

[10] C. Berrou and A. Glavieux, "Near optimum error correcting coding and decoding: turbo-codes," *IEEE Trans. on Communications*, vol. 44, pp. 1261-1271, 1996.

[11] B. Holter, "On the Capacity of the Mimo Channel-A Tutorial Introduction," 2001. Available at
`http://www.iet.ntnu.no/projects/beats/Documents/MIMO_introduction.pdf`,
June 4, 2008.

[12] G. J. Foschini, "Wireless Communication in a Fading Environment When Using Multi-Element Antennas," *Bell Labs Technical Journal*, pp. 41, 1996.

[13] V. Tarokh, N. Seshadri and A. R. Calderbank, "Space-time codes for high data rate wireless communication: performance criterion and code construction," *IEEE Trans. on Information Theory*, vol. 44, pp. 744-765, 1998.

[14] S. M. Alamouti, "A simple transmit diversity technique for wireless communications," *IEEE Trans. on Selected Areas in Communications*, vol. 16, pp. 1451-1458, 1998.

List of Figures

Author

David G. Messerschmitt is the Roger A. Strauch Professor Emeritus of Electrical Engineering and Computer Sciences (EECS) at the University of California at Berkeley, and also a Visiting Professor in the Department of Computer Science and Engineering at the Helsinki University of Technology (HUT). At Berkeley he has previously served as the Interim Dean of the School of Information and Chair of EECS. He is the co-author of five books, including *Digital Communication* (Kluwer Academic Publishers, Third Edition, 2004). He served on the NSF Blue Ribbon Panel on Cyberinfrastructure and co-chaired a National Research Council (NRC) study on the future of information technology research. His doctorate in Computer, Information, and Control Engineering is from the University of Michigan, and he is a Fellow of the IEEE, a Member of the National Academy of Engineering, and a recipient of the IEEE Alexander Graham Bell Medal recognizing "exceptional contributions to the advancement of communication sciences and engineering".