EECS-122 Lectures on Physical Layer

An outline of communication theory and systems

Rahul Jain,
Stavros Tripakis

1 Basic elements of digital communication systems

Digital communications have gained importance and success, mainly because of the advances in electronic circuits and computer science. This fact made information encoded digitally (i.e., in bits) much cheaper to store, manipulate and transmit.

Figure 1 displays the basic building blocks of a digital communication system. Most of these blocks belong to the physical layer of a network. Others can be part of the application layer (e.g., compression). Finally, some blocks may be missing in a computer, for example, the A/D (analog-to-digital) and D/A converters.

2 Signals

The fact that digital communications prevail today does not mean that all signals we might want to transmit are digital. Signals can be classified as:

- Analog: real-valued functions of continuous time (e.g., sound).
- Discrete: real-valued functions of discrete time (e.g., the output of a sensor that measures temperature of a plant every t secs).
- Digital: discrete-valued functions of discrete time (e.g., a binary-encoded image sent on the Internet).

All signals are transformed into digital (and more precisely, binary) signals before being sent through a digital network. This transformation generally involves:

- sampling, to transform an analog signal into a discrete signal, by measuring the value of the signal only at specific time instants;

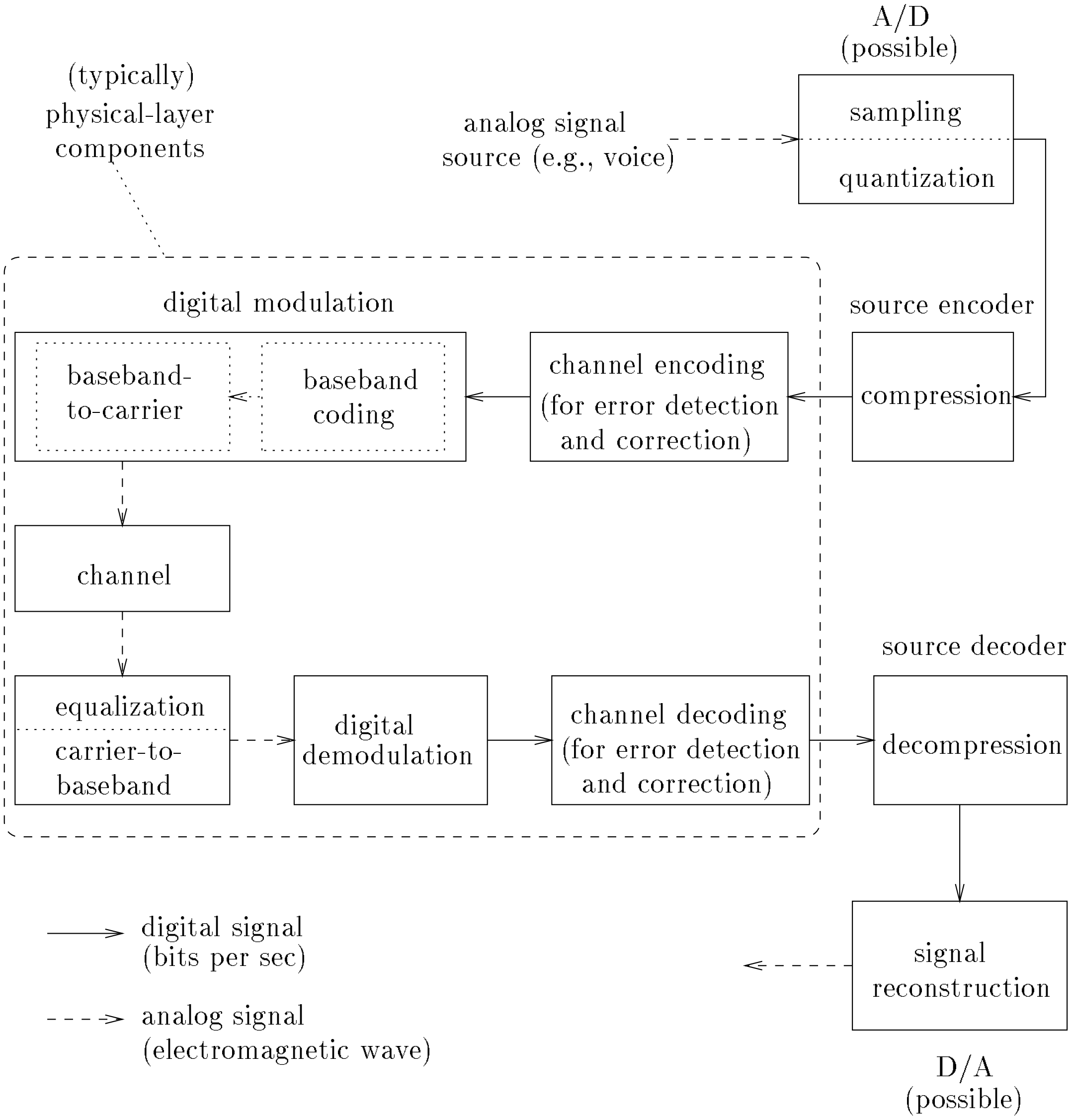

Figure 1: The basic building blocks of a digital communication system.

- quantization, to transform a discrete signal into a digital signal, by encoding the real value by a number of bits.

On the other hand, channels (copper, twisted pair, fiber, vacuum) do not transmit bits. They transmit electromagnetic waves, that is, analog signals. The transformation of a digital signal to an electromagnetic waves and the recovery of the original signal after transmission of the wave through a channel is the role of modulation/demodulation.

Other parts of the picture include encoding/decoding of the digital signal for error detection/correction purposes, and compression of the digital signal for economy purposes.

Before going on, let us define some basic concepts.

Fourier transform (spectrum) of a signal. An analog signal can be written (by Taylor expansion) as a sum (possibly infinite) of sinusoids, each at a different frequency. The spectrum of a signal (its Fourier transform) gives the frequency components of the signal. Given a signal $x(t)$, its Fourier transform is defined as the function

$$X(\omega) = \int_{-\infty}^{\infty} x(t) e^{-j\omega t} dt.$$

Notice that $X(\cdot)$ is defined in the frequency domain whereas $x(\cdot)$ is defined in the (inverse) time domain. Also note that the two representations of the signal are equivalent, since the Fourier transform is invertible: we can construct the signal from its Fourier transform and vice versa.

Bandwidth of a signal. We say that a signal $x(t)$ is band-limited to B Hz if its Fourier transform $X(\omega)$ is zero for all $|\omega| > 2\pi B$. In other words, the maximum frequency component of the signal is B. The bandwidth of a signal band-limited to B Hz is defined to be B Hz.

3 Sampling and Quantization

Sampling and quantization are used when we want to transmit an analog signal through a digital network (e.g., voice over IP).

Sampling. Nyquist proved that a signal which is band-limited to B Hz can be uniquely determined by its values at uniform intervals less than $\frac{1}{2B}$ seconds apart.

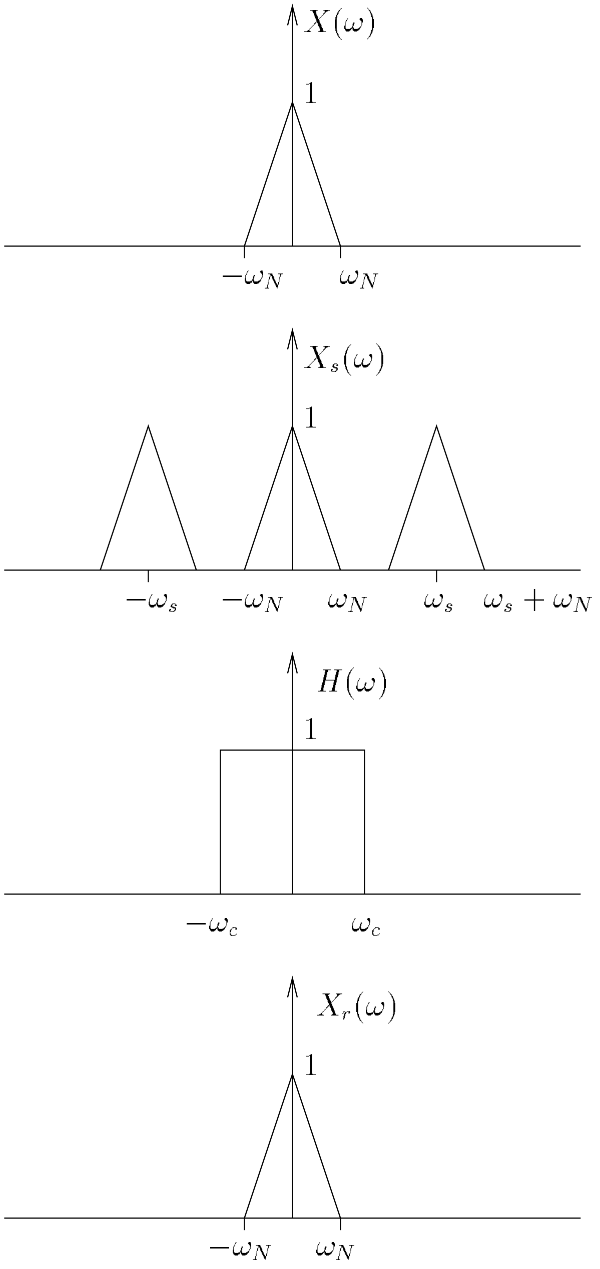

Figure 2: Sampling.

Figure 2 gives the intuition behind the theorem. Let $x(t)$ be the original signal (with $\omega_N = 2\pi B$). Its spectrum $X(\omega)$ is shown in the figure. Let T be the sampling period, that is, the samples are $x(nT)$, for $n \in Z$. Also let $\omega_s = \frac{2\pi}{T}$. The spectrum of the sampled signal is shown second from top. As we can see from the figure, if $\omega_s - \omega_N \leq \omega_N$ then the two curves overlap (aliasing). Therefore, to avoid aliasing, we must have $\omega_s - \omega_N > \omega_N$, or $\omega_s > 2\omega_N$, that is, the sampling frequency must be at least twice the frequency of the signal. Given that $\omega_s > 2\omega_N$, we can reconstruct $x(t)$ by filtering $X_s(\omega)$ with $H(\omega)$, where $\omega_N < \omega_c < \omega_s - \omega_N$. We get $X_r(\omega) = H(\omega)X_s(\omega) = X(\omega)$.

Nyquist's theorem provides the basis for sampling. It implies that, if we want to be able to reconstruct a signal band-limited to B Hz from its samples, then the sampling frequency (how many samples per second) must be greater than $2B$ Hz. For example, if we want to transmit voice over a digital network, we have to sample the voice signal (typically limited to 4 kHz) at least at 8 kHz.

Quantization. Notice that Nyquist's theorem assumes that we can exactly represent each sample (a real value). This is not true in computers, where numbers are encoded as bit-words. Therefore, once we have sampled our signal, we have to represent each sample as a bit-word.

How many bits should we use for each sample? This depends on how accurately we would like to reconstruct the original analog signal. Generally, using n bits per sample means that we divide the range of possible values of the signal into 2^n (disjoint) intervals, and each interval is associated with a different n-bit word. The binary encoding of a sample is the word associated to the interval where the sample lies.

Figure 3 presents an example. Here, $n = 3$. Suppose that the signal takes values in $[0, 1]$. There are 8 quantization intervals, $[0, \frac{1}{8})$, $[\frac{1}{8}, \frac{2}{8})$, and so on. As we see from this example, quantization introduces an error: a sample very close to (but smaller than) $\frac{1}{4}$ would be encoded by 001, as would a sample equal to $\frac{1}{8}$.

In general, the error is proportional to $\frac{1}{2^n}$, when n bits are used per sample. This error can be seen as equivalent to noise added to the signal (called the quantization noise): we suppose that the original signal could only take discrete values $(0, \frac{1}{8}, \frac{2}{8}, ...)$ but some noise (with magnitude smaller than $\frac{1}{2^n}$) is added to each of these values. Since the power is proportional to the square of the magnitude, the noise power is proportional to 2^{-2n}.

The signal-to-(quantization)-noise (SQNR) ratio is defined as the ratio of the power of the signal, to the power of quantization noise. The maximum

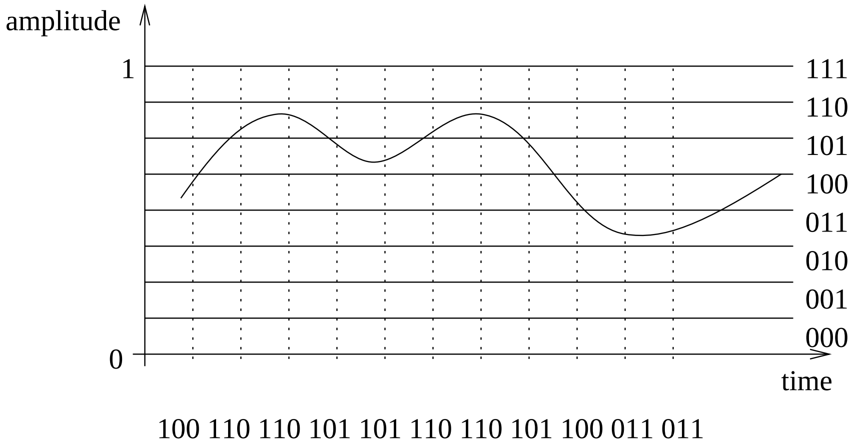

Figure 3: Digitization (sampling and quantization) of an analog signal.

power of the signal here is 1 (since the magnitude of the signal is 1; in general the magnitude could be anything, but we can always normalize it to 1). So, SQNR is proportional to 2^{2n}, which in decibels (dB) is $10\log_{10}(2^{2n}) = 2n \times 10\log_{10}(2) \approx 6n$.

Therefore, given that we want to achieve a SQNR not smaller than α dB, we must use at least $\frac{\alpha}{6}$ bits per sample. Continuing our voice digitization example, if we want a SQNR of 48 dB, we would have to use 8 bits per sample. At a sampling rate of 8 kHz, this yields a digital signal of $8 \times 8 = 64$ kbps.

4 Source Coding

Lossless compression of a file, a picture, or any message means transforming the original message into a new message which is smaller but still contains the same information. Of course, we might want to have lossy compression, where the compressed message does not contain exactly the same information as the original one, but close. In any case, in order to talk precisely about compression, we have to know precisely what we mean by information.

Shannon asked "what is information ?". He realized that information is carried between two parties (a source and a destination) and only when the destination "learns" from the source something it didn't know before. For example, human language contains information because we do not know what somebody will say before he/she starts talking (although we might still not know even after he/she has talked).

Shannon gave an answer by defining the "quantity of information contained in a source" to be what he called the entropy of the source (measured in bits). Let $\{a_1, .., a_n\}$ be the possible messages of the source and assume that message a_i is emitted with probability p_i. The entropy of the source H is defined to be:

$$H = \Sigma_{i=1}^{n} p_i \log_2(\frac{1}{p_i}).$$

For example, a source with two messages a, b which are emitted each with probability 0.5 has entropy $2 \cdot 0.5 \cdot \log_2(2) = 1$ bit. Indeed, using one bit we can encode the message a or b. Notice that a source that can transmit a single message with probability 1 has zero entropy, that is, contains no information (since there is no uncertainty as to what is the source going to say).

Theorem 1 (Shannon's source coding theorem) *Let X be the message from a discrete memoryless source with finite entropy $H(X)$. Blocks of J symbols from the source are encoded into code words of length N from a binary alphabet. Then, from any $\epsilon > 0$, the probability P_e of a block decoding failure can be made arbitrarily small if*

$$R = \frac{N}{J} \geq H(X) + \epsilon,$$

where J is sufficiently large. Conversely, if

$$R \leq H(X) - \epsilon$$

then $P_e \rightarrow 1$ as J is made sufficiently large. In fact, $H(X)$ is achievable.

The theorem says that a message X with entropy $H(X)$ cannot be compressed into less than $H(X)$ bits without loss of information, and this lower bound is achievable (e.g., Huffman codes achieve the bound). The theorem provides the basis for source codes, used in data compression. We will study compression later in the course. For the moment, let us just mention that source codes include Huffman codes (optimal codes), run-length codes (e.g. Lempel-Ziv), and so on.

5 Communication Channels

As a signal passes through a channel, it gets modified and its characteristics get changed. The main effects are:

- Attenuation: loss in energy of the electromagnetic (EM) wave as it propagates due to absorption and dissipation. This results in decrease in signal amplitude.

- Distortion: different frequency components of a waveform may be attenuated and delayed differently (propagation speed depends on frequency), and this may result in distortion in shape of the waveform.

- Dispersion: as a burst of EM energy propagates, it spreads.

- Noise: There is thermal noise and interference in the communication channel that adds to the signal. (Note that noise can be multiplicative but we usually only worry about additive noise).

In general, the channel can be viewed as a function which can be nonlinear and may have memory (e.g., in wireless channels with multi-path). Assuming the channel is memoryless, we can represent the received signal $y(t)$ as $h(x(t))$, where $x(t)$ is the transmitted signal, and $h(\cdot)$ models the channel. By expanding $h(x(t))$, we have in general: $y(t) = f(x(t)) = a_0 + a_1 x(t) + a_2 x(t)^2 + \cdots$.

If we assume the channel to be linear, we have $Y(\omega) = H(\omega)X(\omega)$, where $Y(\omega)$ and $X(\omega)$ are the Fourier transforms of $Y(t)$ and $x(t)$ and $H(\omega)$ is the channel response.

Bandwidth of a channel. A cut-off channel does not allow all frequencies to propagate through it. The bandwidth of the channel is defined to be B (in Hz) if $H(\omega) \neq 0$ only if $\omega \in [-2\pi B, 2\pi B]$.

The channel also adds noise. We usually assume AWGN (additive white Gaussian noise), such that $y(t) = h(t) \star x(t) + n(t)$, where $n(t)$ models the noise.

Bandwidth and noise are the two factors that limit the capacity of a channel (how many bits per second the channel can carry). The precise relation between capacity, bandwidth and noise is stated by Shannon's channel coding theorem (see below). As it follows from this theorem, without noise, we could transmit at arbitrarily large bit rate, independently of the bandwidth of the channel.

Equalization. Equalization is used by the receiver to undo the distortion, attenuation and dispersion effects introduced by the channel. The idea is the following. If we know the channel's response, $H(\omega)$, then we introduce an equalization filter with spectrum $E(\omega) = \frac{1}{H(\omega)}$. In practice, $H(\omega)$ is changing dynamically, so equalizers must adapt to these changes. Types of such equalizers are so-called decision-feedback or zero-forcing equalizers.

6 Channel Coding

Shannon asked the following question: "how many bits per second can we transmit (without errors) through a channel, given the channel's bandwidth and noise ?". Shannon answered the question by proving the so-called Shannon's channel coding theorem. This theorem relates the capacity of a channel (i.e., how many bits per second we can transmit) with its bandwidth and noise.

Theorem 2 (Shannon's channel coding theorem) *Given a channel of bandwidth B and signal-to-noise ratio (SNR) γ, we can transmit at R bps over the channel with bit error rate approaching zero, iff $R \leq C$, where*

$$C = B \log_2(1+\gamma).$$

In the above formula, C is in `bits/s`, B is in `Hz` (or `1/s`), and the logarithm is in `bits`. The SNR γ is **not** in dB (decibels) but is unitless (for example, if the SNR is 20 dB, then $\gamma = 10^{\frac{20}{10}} = 100$).

Shannon's channel coding theorem gives a fundamental limit. No clever tricks can beat it! In fact, the upper bound C is achievable, that is, Shannon's proof gives a coding scheme (a block code) that achieves bit-rate C.

Notice that, in practice, other error detection/correction codes are used (c.f. lecture 24), such as Hamming codes, CRC, convolutional codes, Turbo codes, and so on. Why don't we use Shannon's code (the one in the proof), which achieves the maximum capacity, instead ? This is because this code cannot be computed in real-time (since it is defined over the entire message), therefore, it is not practical. Efficient codes such as those mentioned above have been developed since Shannon's results. All these codes obey Shannon's theorem.

7 Modulation

As mentioned already, channels do not carry bits, they carry electromagnetic waves. For instance, an EM wave propagates along a twisted pair cable when we vary the voltage V applied to the ends of the cable over time (changes in voltage result in changes in the electric current through the cable, which generates the EM wave). Modulation is used to transform a bit-stream into an analog signal (e.g., $V(t)$). In general, we talk about baseband modulation if the bandwidth of the resulting analog signal is approximately the same as the "frequency" (rate) of the bit-stream. Manchester encoding (see below) used in Ethernets is an example of such a modulation. We talk about broadband modulation when the analog signal is in a different (typically higher) frequency band. For example, wireless LANs transmit at rates in the order of Mbps (10^6 bps) but at frequencies in the order of GHz (10^9 Hz).

7.1 Baseband Modulation

In the sequel, we consider an input bit-stream at rate R bps and let $T = \frac{1}{R}$. So, one bit gets delivered at the modulator input every T seconds.

Polar encoding. The idea is to represent, say, bit 1 by +1 volt and bit 0 by −1 volt. So, for $n = 0, 1, ...,$ and $nT \leq t < (n+1)T$, the output analog signal is defined as:

$$x(t) = \begin{cases} 1 & \text{if the } n\text{-th bit is 1} \\ -1 & \text{if the } n\text{-th bit is 0} \end{cases}$$

Notice that the maximum frequency of the output signal is R Hz and it corresponds to the input bit-stream 1010101010·.

The problem with polar modulation is that a long sequence of 1s gives rise to a signal which remains constant for a long time, which results in de-synchronization problems (see below). This is why self-synchronizing schemes have been proposed, such as those described below.

Manchester encoding. The idea is to represent bit 1 by a "falling edge" and bit 0 by a "rising edge", as shown in figure 4 (this is figure 7.8 from Walrand's book). So, for $n = 0, 1, ...,$ the output analog signal is defined in this case as:

$$x(t) = \begin{cases} 1 & \text{if the } n\text{-th bit is 1 and } nT \leq t < nT + \frac{T}{2} \\ 0 & \text{if the } n\text{-th bit is 1 and } nT + \frac{T}{2} \leq t < (n+1)T \\ 0 & \text{if the } n\text{-th bit is 0 and } nT \leq t < nT + \frac{T}{2} \\ 1 & \text{if the } n\text{-th bit is 0 and } nT + \frac{T}{2} \leq t < (n+1)T \end{cases}$$

This scheme is guaranteed to generate one transition of the signal per bit period T, which helps synchronization of the sender's and receiver's clocks. A disadvantage of this scheme is that the output signal has maximum frequency (thus bandwidth) $2R$, that is, double the rate of the input bit-stream. This motivates the following scheme.

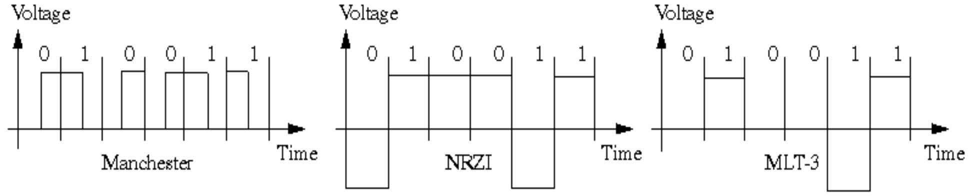

Figure 4: Some widely used self-synchronizing baseband modulation schemes.

NRZI encoding. Contrary to the above schemes, the non-return to zero with inversion (NRZI) scheme has memory. That is, the value of the output at some time generally depends not only on the input bit, but also on the value of the output in the past. But the idea is really simple: initially, output, say, $+1$ volt for a bit 1, and -1 for 0. For each subsequent bit, if this bit is 0 then output the same voltage as in the previous bit-period, otherwise output the negation of the previous voltage.

This scheme generates less transitions than the Manchester coding, in fact, its maximum frequency is R. However, a long sequence of 0s generates no transitions, so it presents the same de-synchronization problem as polar coding. For this reason, NRZI is almost always used with a pre-coding of the bit-stream, called 4B/5B, where four bits are replaced by five (according to a table) so that the resulting bit-stream has at most 2 consecutive 0s. So, after 4B/5B pre-coding, the bit-stream has rate $\frac{5}{4}R$.

Pulse shaping. In reality, bits are not sent as rectangular pulses, because in the frequency domain such pulses are not band-limited and cause inter-symbol interference (ISI). Similarly, if the pulse shape is band-limited, it causes aliasing in the time domain and again there is ISI. According to Nyquist, if a pulse satisfies three conditions (called Nyquist's criteria) then at the points of sampling the ISI is zero.

7.2 Broadband Modulation

Amplitude Shift Keying (ASK). The idea is to represent a bit 0 by a voltage of 0 and bit 1 by a sinusoid of some frequency f. So, for $n = 0, 1, ...,$ and $nT \leq t < (n+1)T$, the output analog signal is defined as:

$$x(t) = \begin{cases} p(t) \cdot \sin(2\pi f t) & \text{if the } n\text{-th bit is 1} \\ 0 & \text{if the } n\text{-th bit is 0} \end{cases}$$

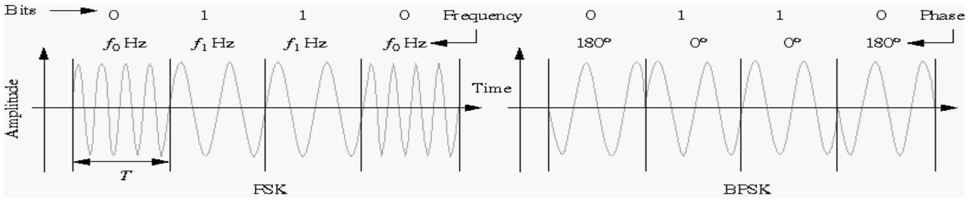

Figure 5: Frequency Shift Keying and Phase Shift Keying.

where p is some function of time.

Frequency Shift Keying (FSK). The idea is to represent a bit 0 by a sinusoid of some frequency f_0 and bit 1 by a sinusoid of some frequency f_1. So, for $n = 0, 1, ...,$ and $nT \leq t < (n+1)T$, the output analog signal is defined as:

$$x(t) = \begin{cases} \sin(2\pi f_1 t + \theta) & \text{if the } n\text{-th bit is 1} \\ \sin(2\pi f_0 t + \theta) & \text{if the } n\text{-th bit is 0} \end{cases}$$

where θ is some phase. Figure 5 shows an example.

Phase Shift Keying (PSK). The idea here is to change phases instead of frequencies. So, for $n = 0, 1, ...,$ and $nT \leq t < (n+1)T$, the output analog signal is defined as:

$$x(t) = \begin{cases} \sin(2\pi f t + \theta_1) & \text{if the } n\text{-th bit is 1} \\ \sin(2\pi f t + \theta_0) & \text{if the } n\text{-th bit is 0} \end{cases}$$

where θ_0, θ_1 is the phases for bit 0 and bit 1, respectively, and f is the frequency of the sinusoid. Figure 5 shows an example, where $\theta_0 = 0$ and $\theta_1 = \pi$.

Quadrature Amplitude Modulation (QAM). This is a generalization and adaptation to the digital case of Amplitude Modulation (AM) of analog signals (see below), which is used in AM radio. In AM, the output signal is $y(t) = x(t)\ \sin(2\pi f t)$, where $x(t)$ is the input signal.

In QAM, for $i = 0, 1, ...,$ and $iT \leq t < (i+1)T$, the output signal is

$$y(t) = a_i \cos(2\pi f t) + b_i \sin(2\pi f t)$$

where a_i and b_i depend on the i-th input bit.

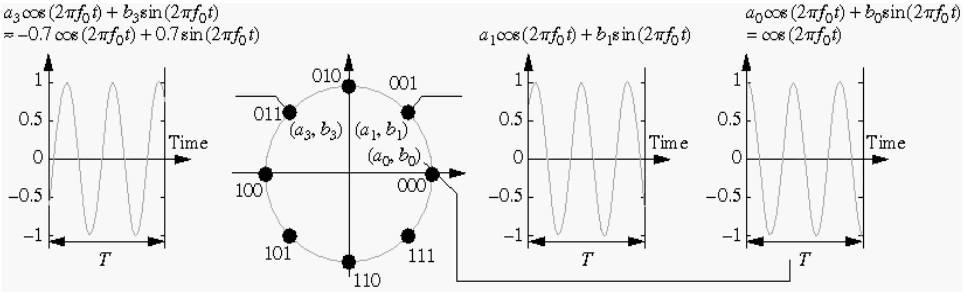
$a_3 \cos(2\pi f_0 t) + b_3 \sin(2\pi f_0 t)$
$\approx -0.7 \cos(2\pi f_0 t) + 0.7 \sin(2\pi f_0 t)$
$a_1 \cos(2\pi f_0 t) + b_1 \sin(2\pi f_0 t)$
$a_0 \cos(2\pi f_0 t) + b_0 \sin(2\pi f_0 t)$
$= \cos(2\pi f_0 t)$
1
0.5
0
−0.5
−1
Time
T
010
001
011
(a_3, b_3)
(a_1, b_1)
(a_0, b_0)
100
000
101
111
110

Figure 6: Quadrature Amplitude Modulation (QAM).

In practice, the input bits are not taken one-by-one, but are grouped in groups of k bits, and the coefficients of $\cos(2\pi ft)$ and $\sin(2\pi ft)$ change only once every kT seconds.

Figure 6 shows an example (read $3T$ instead of T).

7.3 Analog Modulation

As mentioned above, modulation is used also in analog communications, such as radio transmissions. Here we discuss the most frequent analog modulation schemes. In the following, $x(t)$ is the input (analog) signal and $y(t)$ is the output (analog) signal (after the modulation).

Amplitude modulation (AM). In AM, the output signal is $y(t) = x(t) \cdot \sin(2\pi ft)$. If $X(\omega)$ is the Fourier transform of $x(t)$ then the Fourier transform of $y(t)$ is $Y(\omega) = \frac{1}{2}(X(\omega + \omega_c) + X(\omega - \omega_c))$, where $\omega_c = 2\pi f$.

Frequency modulation (FM) and Phase modulation (PM). In FM, the output signal is

$$y(t) = A \, \cos(\omega_c t + k_f \int_{-\infty}^{t} x(s) ds)$$

where A, ω_c and k_f are constants.

In PM, the output signal is

$$y(t) = A \, \cos(\omega_c t + k_p x(t))$$

where A, ω_c and k_p are constants.

FM and PM are non-linear modulations. Although quite similar, FM and PM do have differences, for instance, PM can have jump discontinuities, whereas FM does not.

8 Synchronization

The transmitter and receiver use clocks (or oscillators), for modulating the (digital) signal to be transmitted and demodulating the (analog) signal received (after equalization and pulse shaping), respectively. These clocks are not perfect, that is, they do not "tick" at a constant rate (in particular, the two clocks might not tick at exactly the same rate). Synchronization techniques are used to adapt the clock of the receiver to the rate of the clock of the transmitter. Here we present one such technique, called phase locked loop.

Let $x(t)$ be the (analog) signal output by the channel. The signal $x(t)$ is the result of addition of noise and other effects (distortion, dispersion, attenuation) introduced by the channel. Here, we ignore all the rest and take $x(t)$ to be:

$$x(t) = \sin\left(2\pi f_0 t + \phi(t)\right).$$

In the above formula, $\phi(t)$ is used to model both the changes in the rate of the transmitter, but also the phase shift due to the propagation delay introduced by the channel: if $z(t)$ is what the transmitter put into the channel and the propagation delay is δ, then $x(t+\delta) = z(t)$. Notice that the receiver does not know $\phi(t)$ (in fact, the receiver is trying to reconstruct $x(t)$ by sampling the channel).

Let $y(t) = \cos\left(2\pi f(t) t\right)$ be the clock of the receiver. We assume that the clock is a controllable device, with output $y(t)$ and some input a, such that the rate of the clock $f(t)$ slowly increases when $a > 0$ and slowly decreases when $a < 0$. Equivalently, we can write $y(t)$ as $y(t) = \cos\left(2\pi f_0 t + \theta(t)\right)$. (Changing the rate $f(t)$ is equivalent to changing the phase $\theta(t)$, since $\cos(2\pi f(t)) = \cos(2\pi f_0 + \theta(t))$, for $\theta(t) = 2\pi \frac{f(t)}{f_0}$.

The phase locked loop technique is based on the following idea: can we keep the phase of the receiver's clock $(2\pi f_0 t + \theta(t))$ close to the phase of the signal $(2\pi f_0 t + \phi(t))$, for all t ? If we can do that, then the receiver knows when to sample the channel (for instance, it might decide to sample whenever the phase of the clock is 0). This comes down to keeping $\theta(t)$ close to $\phi(t)$, for all t.

To do that, we define the input a of the receiver's clock to be a function

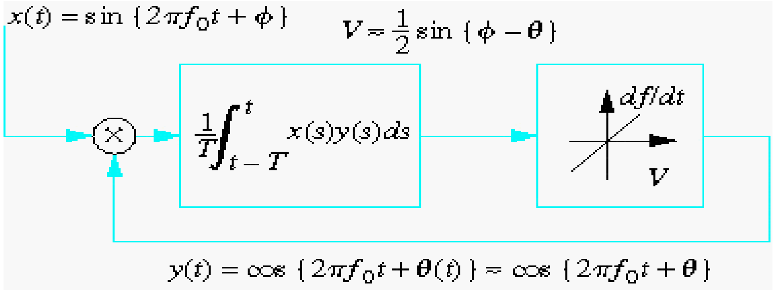
$x(t) = \sin\{2\pi f_0 t + \phi\}$
$V \approx \frac{1}{2}\sin\{\phi - \theta\}$
$\frac{1}{T}\int_{t-T}^{t} x(s)y(s)ds$
df/dt
V
$y(t) = \cos\{2\pi f_0 t + \theta(t)\} \approx \cos\{2\pi f_0 t + \theta\}$

Figure 7: Phase-locked loop.

of time:

$$a(t) = \frac{1}{T}\int_{t-T}^{t} x(s) \cdot y(s) ds$$

where T is some time interval over which we average the product $x(s)y(s)$.

Assuming that $\theta(t) \approx \phi(t)$, we get:

$$a(t) \approx \frac{1}{2}\sin(\phi(t) - \theta(t)).$$

This kind of feedback control (illustrated in figure 7) is stable and makes $\theta(t)$ converge to $\phi(t)$. Indeed, if $\theta(t)$ is slightly smaller than $\phi(t)$, then $a(t) > 0$ and the rate $f(t)$ increases, which is equivalent to $\theta(t)$ increasing (remember, $\theta(t)$ is nothing but $2\pi\frac{f(t)}{f_0}$). Similarly, if $\theta(t)$ is slightly greater than $\phi(t)$, then $a(t) < 0$, which makes $\theta(t)$ decrease. The circuit of figure 7 would typically be part of the demodulation block of the receiver.

9 Multiple Access

Some channels (e.g., wireless) are not (and cannot be) isolated, so the problem arises, how can multiple users share the same channel ? Apart from higher level protocols (at the medium-access control – MAC – level), techniques directly at the physical layer can also be used. Much of the discussion of this section is particularly relevant today, with the renewed interest in wireless digital communications.

9.1 Time division multiple access (TDMA)

TDMA is based on time division multiplexing: allow each user to access the resource (in this case, the channel) only during a specific time interval (a time slot). For example, if we have 10 users, time can be divided into 10 slots, and user 1 transmits during the first slot, user 2 during the second slot, and so on.

This technique is still used in cellular telephony. Problems include:

- Clock synchronization: the clocks of all users do not work at exactly the same rate, so one user might start transmitting before another user has finished, which results in collisions.

- (Efficient) resource allocation: how "long" should the time slots be ? should they all have the same size ? how are they allocated to the users ? One user might need more slots than another, but the needs also vary in time. Therefore, having fixed-length slots allocated permanently to users might be too wasteful, since many slots will be left idle.

- ISI (inter-symbol interference): pulses which are limited in the time domain (here, limited in a certain slot) result in aliasing in the frequency domain, therefore, ISI. To reduce ISI, slots are "guarded" at their ends, for instance, if the time slot is $[n\delta, (n+1)\delta]$, then the user only transmits during $[n\delta+\epsilon, (n+1)\delta-\epsilon]$, which reduces the effective bandwidth.

9.2 Frequency division multiple access (FDMA)

Here, instead of dividing time, we divide frequency. Assuming that the channel's bandwidth is B (so, the frequency band is $[f-B, f+B]$), we divide the band into sub-bands, $[f-B, f-B+B_1]$, $[f-B+B_1, f-B+B_1+B_2]$, and so on. The first sub-band has (a smaller than B) bandwidth B_1, the second one has bandwidth B_2, and so on. Each user is only transmitting in one of the sub-bands.

ISI is also a problem in FDMA: here the signal is band-limited in the frequency domain, which results in aliasing in the time domain. A similar technique as in TDMA is used to reduce ISI, by "guarding" the sub-bands at their ends. So, instead of using the entire sub-band $[f-B, f-B+B_1]$, a transmitter uses only $[f-B+\epsilon, f-B+B_1-\epsilon]$.

Also, in practice, the number of users is much higher than the number of sub-bands. This creates a problem of resource allocation: how does a user choose which sub-band to use so that interference with other users is minimized ?

9.3 Code division multiple access (CDMA)

In CDMA, all users transmit at the same time and use the whole available bandwidth of the channel, however, they use different “codes”. Depending on the scheme used (whether DSSS or FHSS, see below) a code means different things. In any case, the objective is to keep these codes invertible (a code c_1 in some algebra is invertible if $c_1 \cdot c_1 = 1$) and orthogonal (two codes c_1 and c_2 are orthogonal if $c_1 \cdot c_2 = 0$), so that more than one users transmitting simultaneously at different codes will not interfere with each other.

For example, suppose two transmitters send messages M_1 and M_2, encoded as $M_1 \cdot c_1$ and $M_2 \cdot c_2$, respectively. A receiver within range of both transmitters gets the signal $M_1 \cdot c_1 + M_2 \cdot c_2$. If this receiver is “listening” to the code c_1, then it decodes the signal by computing: $(M_1 \cdot c_1 + M_2 \cdot c_2) \cdot c_1 = M_1 \cdot c_1 \cdot c_1 + M_2 \cdot c_2 \cdot c_1 = M_1 + 0 = M_1$, which is the message the receiver was intended to get (thus, interference from the second transmitter is eliminated).

CDMA techniques are called spread spectrum techniques, because they result in the bandwidth required to transmit the data (approximately of the same order of magnitude as the bit-rate of the transmission) being many orders of magnitude smaller than the bandwidth of the channel. That is, the bandwidth of the original data signal “spreads” and its energy decreases, so that the resulting signal appears as white noise.

Direct Sequence Spread Spectrum (DSSS). The idea behind DSSS is that each user encodes its bit-stream using a special code bit sequence, called a chip. Using a bit of length n results in each bit of the original data stream to be replaced by a sequence of n bits. So, if the rate of the original stream is R bps, the rate of the encoded stream is nR bps: thus, with DSSS, the bandwidth of the signal “spreads” (is multiplied) by n and the energy is divided by n.

More precisely, if $c_1 c_2 \cdots c_n$ is the chip and $b_0 b_1 \cdots$ is the data bit-word, then the output signal is:

$$e_0^1 e_0^2 \cdots e_0^n e_1^1 e_1^2 \cdots e_1^n \cdots,$$

where $e_i^j = b_i \cdot c_j$. For example, assume that the chip is: $1\ {-1}\ 1\ 1$, and the data bit word is: 100111010. Then, the resulting sequence is:

$$1\ {-1}\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ {-1}\ 1\ 1\ 1\ {-1}\ 1\ 1\ 1\ {-1}\ 1\ 1\ 0\ 0\ 0\ 0\ 1\ {-1}\ 1\ 1\ 0\ 0\ 0\ 0$$

Frequency Hopping Spread Spectrum (FHSS). The idea behind FHSS is to split the original message into uniform size chunks and transmit each chunk on a different frequency. The sequence of frequencies used by each user is pseudo-random, so that different sequences have small probability of interference. It should be noted that FHSS techniques have originally been developed in the military, because the provide protection against so-called "jammers" (trying to disable or intercept the transmission).

10 Special Problems in Wireless Channels

Wireless channels are difficult, mainly because they cannot be isolated. This means that the medium is multiple-access, which implies that it is very vulnerable to interference.

Frequency bands. The multiple-access nature of wireless channels also results in frequency bands being scarce resources. Most frequency bands in most places in the world are controlled by the government (frequency bands are used for radio and TV broadcasting, satellite transmissions, military things, etc).

So-called unlicensed bands are available for use by individuals and companies, however, not the same bands are available all over the world. This results in products (e.g., cellular phones) being incompatible, because they do not operate in the same bands (cellular phones are incompatible also because they do not use the same multiple-access protocols). Unlicensed bands in the US include the 900 MHz, 2.4 GHz and 5.8 GHz ISM bands.

Infrared transmission uses much higher frequency bands than radio (infrared is typically around 100 GHz). The advantages are that these bands are unlicensed, and they do not suffer from interference from radio. The main problem is due to the fact that the power of the signal decreases proportionally to the square of the frequency of the signal (see below). In frequencies as high as infrared, this means that infrared transmission is limited to very short distances (in the order of tens of meters). A potential solution is to use directional antennas, however, they suffer from fading due to obstacles (see below).

We now discuss other problems of wireless channels.

Path loss (or path fading). Path loss is another name for attenuation of the power of the transmitted signal as the distance from the transmitter increases. An important thing to note is that path loss is higher at higher frequencies. If P is the transmitted signal power, f is the frequency of the signal and d is the distance between the transmitter and the receiver, then the power of the signal at the receiver is proportional to $\frac{P}{f^2 \cdot d^\alpha}$, where α is a constant greater than one (typically, α is between 2 and 4).

Shadow fading (or shadowing). Shadowing refers to the attenuation of the power of the signal due to obstacles (e.g., buildings).

Multi-path fading (or multi-path). Multi-path happens when the receiver gets multiple "copies" of the transmitted signal, each with a difference phase. The "copies" have different phase because they have potentially traveled a different distance (due to reflections). The copies are added-up at the receiver and may cancel each other. In general, they result in a signal whose power varies greatly with time. In fact, multi-path is perhaps the harder problem in wireless communications.

Interference. Interference in wireless networks comes from the limited amount of frequencies and the fact that the medium is not isolated. Since there are many users and not enough frequencies, frequencies have to be reused. Two or more users transmitting at approximately the same frequencies will interfere with each other, if they are not too far apart (path-loss practically limits the range of a transmitter).

Due to the above reasons, wireless channels present a much higher bit-error rate (BER) than wirelined channels (typical values for BER in wireless channels are in the order of 10^{-3}).

Cellular phones, palm-tops and wireless transmission technologies. The field of wireless communications is currently very active, and new technologies and products emerge very rapidly. Research is done to determine optimal transmission policies, ways to dynamically control transmission power and other parameters (hopping frequencies, codes, etc) in order to minimize interference, medium-access protocols, and so on. New products become available that use one or combinations of the above techniques (e.g., some cellular phones use a combination of TDMA/FDMA, others – such as the European GSM – use TDMA with FHSS). It remains to be seen which of the emerging technologies will prevail and how they will evolve.

References

[1] B. P. Lathi. “Modern communication systems”, Wiley, 1968.

[2] J. G. Proakis. “Digital communications”, 3rd edition, McGraw-Hill, 1995.

[3] T. Rappaport. “Wireless communications: principles and practice”, Prentice-Hall, 1996.

[4] J. Walrand. “Communication networks: a first course”, 2nd edition, McGraw-Hill, 1998.

[5] J. Walrand and P. Varaiya. “High-performance communication networks”, 2nd edition, Morgan-Kaufmann, 1999.