ECE 5325/6325: Wireless Communication Systems
Lecture Notes, Spring 2013

Lecture 12

Today: (1) Intro to Digital Communications

- HW 5 due Thursday at the start of class.
- Reading – Today: Molisch 11, MUSE video. Thu: Molisch 12.1.

1 Digital Communications: Overview

My six word summary of digital communications: **Use linear combinations of orthogonal waveforms**.

1.1 Orthogonal Waveforms

My "engineering" definition of a set of orthogonal waveforms: They are waveforms that can be separated at the receiver.

Def'n: *Orthogonal*
Two real-valued waveforms (finite-energy functions) $\phi_1(t)$ and $\phi_2(t)$ are *orthogonal* if

$$\int_{-\infty}^{\infty} \phi_1(t)\phi_2(t)dt = 0$$

Two complex-valued waveforms $\phi_1(t)$ and $\phi_2(t)$ are orthogonal if

$$\int_{-\infty}^{\infty} \phi_1(t)y^*(t)dt = 0$$

where $y^*(t)$ is the complex conjugate of $\phi_2(t)$.

Def'n: *Orthogonal Set*
N waveforms $\phi_1(t), \ldots, \phi_N(t)$ are mutually orthogonal, or form an orthogonal set, if every pair of waveforms $\phi_i(t)$, $\phi_j(t)$, for $i \neq j$, is orthogonal.

Example: Sine and Cosine
Let

$$\begin{aligned}
\phi_1(t) &= \begin{cases} \cos(2\pi t), & 0 < t \leq 1 \\ 0, & o.w. \end{cases} \\
\phi_2(t) &= \begin{cases} \sin(2\pi t), & 0 < t \leq 1 \\ 0, & o.w. \end{cases}
\end{aligned}$$

Are $\phi_1(t)$ and $\phi_2(t)$ orthogonal?

Example: Frequency Shift Keying
Assume $T_s >> 1/f_c$, and show that these two are orthogonal.

$$\begin{aligned}\phi_0(t) &= \begin{cases} \cos(2\pi f_c t), & 0 \leq t \leq T_s \\ 0, & o.w. \end{cases} \\ \phi_1(t) &= \begin{cases} \cos\left(2\pi \left[f_c + \frac{1}{T_s}\right] t\right), & 0 \leq t \leq T_s \\ 0, & o.w. \end{cases}\end{aligned}$$

1.2 Linear Combinations

What is a linear combination of orthogonal waveforms? Well, consider the orthogonal set $\phi_1(t), \ldots, \phi_N(t)$. A linear combination $s_i(t)$ is

$$s_i(t) = a_{i,1}\phi_1(t) + a_{i,2}\phi_2(t) + \cdots + a_{i,N}\phi_N(t) = \sum_{i=k}^{N} a_{i,k}\phi_k(t)$$

We also call the a linear combination a *symbol*. We use subscript i to indicate that it's not the only possible linear combination (or symbol). In fact, we will use M different symbols, so $i = 1, \ldots, M$, and we will use $s_1(t), \ldots, s_M(t)$.

We represent the ith symbol (linear combination of the orthogonal waveforms), $s_i(t)$, as a vector for ease of notation:

$$\mathbf{a}_i = [a_{i,1}, a_{i,2}, \ldots, a_{i,N}]^T$$

The superscript T is for transpose – $\mathbf{a}_i$ is a column vector. Vectors are easy to deal with because they can be plotted in vector space, to show graphically what is going on. We call the plot of all possible $\mathbf{a}_i$, that is, for $i = 1, \ldots M$, the *constellation diagram*. Some examples are shown in Figure 1.

1.3 Using M Different Linear Combinations

Here is how a transmitter *uses* the different linear combinations to convey digital bits to the receiver. First, consider that there are M different symbols for the TX to chose from. Each symbol is described by a $\log_2 M$-length bit sequence. For example, if there are 8 possible combinations, we would label them 000, 001, 011, 010, 110, 111, 101, 100.

The transmitter knows which $\log_2 M$-bit sequence it wants to send. It picks the symbol that corresponds to that bit sequence, let's call it symbol i. Then it sends $s_i(t)$.

If the receiver is able to determine that symbol i was sent, it will correctly receive those $\log_2 M$ bits of information. In this example, it will receive three bits of information.

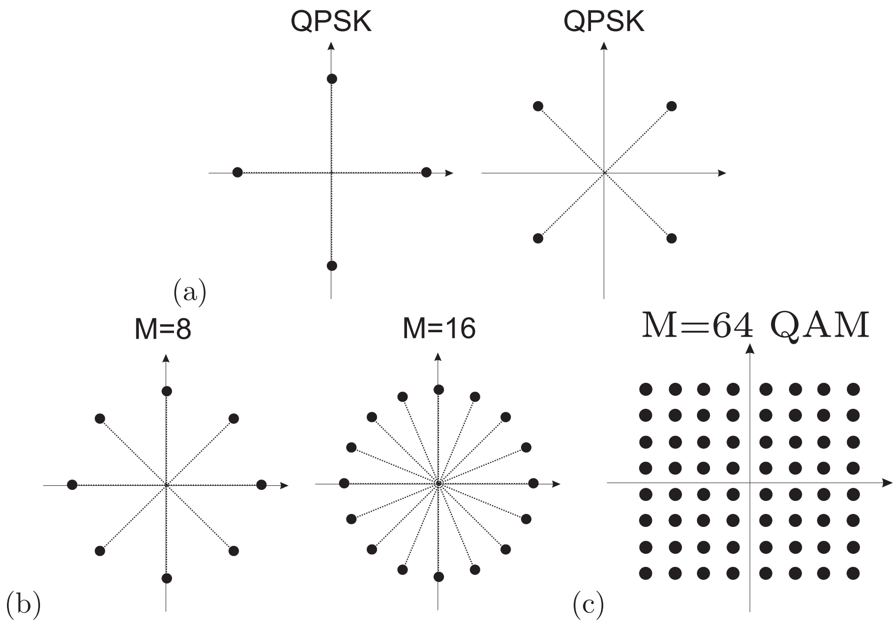

Figure 1: Signal constellations for (a) $M = 4$ PSK (a.k.a. BPSK), (b) $M = 8$ and $M = 16$ PSK, and (c) 64-QAM.

1.4 Reception

At a receiver, because we've used an orthogonal set, we can determine how much of each waveform was sent. This can be done using a bank of matched filters. In short, we can recover an attenuated $\mathbf{a}_i$ plus noise. We might write $\mathbf{x} = g\mathbf{a}_i + n_i$ where g is the total gain ($g \ll 1$) introduced by the channel and antennas, and n_i is the random additive noise added in at the receiver. Assume we have receiver gain that multiplies the received filter by $1/g$ (to cancel out g). If the resulting $\hat{\mathbf{a}}_i = \mathbf{x}/g$ is close enough to the actual transmitted signal $\mathbf{a}_i$, then the correct bit string is received.

How does the receiver compute the $\hat{\mathbf{a}}_i$? It does this using a matched filter receiver. A matched filter receiver has one bank for each orthogonal waveform (the $\phi_1(t), \ldots, \phi_N(t)$. Bank k has a filter that quantifies how much of $\phi_k(t)$ is present in the received signal. Because the waveforms are orthogonal, the job is simple. An analogy would be to a really good division of labor between members of a team – filter k comes up with the answer for waveform k, without being distracted by the other waveforms which may or may not be present in the received signal.

1.5 How to Choose a Modulation

A *digital modulation* is the choice of: (1) the linear combinations $\mathbf{a}_1, \ldots, \mathbf{a}_M$ and, (2) the orthogonal waveforms $\phi_1(t), \ldots, \phi_N(t)$. We will choose a digital modulation as a tradeoff between the following characteristics:

1. Bandwidth efficiency: How many bits per second (bps) can be sent

per Hertz of signal bandwidth. Thus the bandwidth efficiency has units of bps/Hz.

2. Power efficiency: How much energy per bit is required at the receiver in order to achieve a desired fidelity (low bit error rate). We typically use S/N or E_s/N_0 or E_b/N_0 as our figure of merit.

3. Cost of implementation: Things like symbol and carrier synchronization, and linear transmitters, require additional device cost, which might be unacceptable in a particular system.