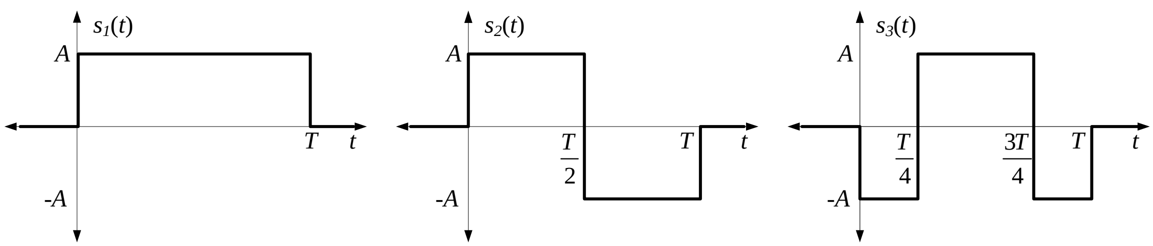
EE 1473 - Digital Communication Systems
Homework Set 7

Due: Thursday, March 21, 2019

1. In this problem, we consider a *ternary* communication system in which one of 3 symbols is transmitted every T seconds using the signals shown in the figure below. When symbol $i \in \{1, 2, 3\}$ is transmitted, the received signal is

$$H_i: \quad r(t) = s_i(t) + n(t),$$

where $n(t)$ is a white Gaussian noise process with mean 0 and power spectral density $P_n(f) = N_0/2$. Each symbol is equally likely to be transmitted, so the priors are $P_i = 1/3$.



   (a) Show that the signals $s_i(t)$ are mutually orthogonal.

   (b) Determine signals $\phi_i(t)$ that are versions of the $s_i(t)$ that have been normalized to unit energy.

   (c) Sketch a block diagram of the optimal receiver for this problem, as a bank of 3 correlation receivers, followed by a decision rule. Each channel should correlate with one of the energy-normalized signals $\phi_i(t)$ from part (b).

   Let r_i be the output of channel i in your receiver. Specify the decision rule that takes $\{r_1, r_2, r_3\}$ as inputs and determines which signal is most likely to have been transmitted. Then simplify the decision rule to comparing pairwise differences of the r_i to thresholds.

   (d) Determine the signal constellation for this system, in terms of A and T.

   (e) Express the probability that a symbol will be decoded in error, in terms of differences between the receiver outputs, as defined in part (c). Then explain why it is difficult to express the symbol error probability in terms of the Gaussian cdf.

   (f) Let the symbol rate be $D = 1.544$ Mbaud and $A = 2$ Volts. Use MATLAB to plot the signal constellation; what are the units for the axes? Since the decision space is three-dimensional, use the `plot3` command in MATLAB to do this.

2. In this problem, we consider a modification of the system from Problem 1, in which one of 8 symbols is transmitted every T seconds, using the signals

$$y_1(t),\, y_2(t),\, \ldots,\, y_8(t).$$

Each of the 8 signals is equally likely to be transmitted.

The system must conform to the following specifications:

- The receiver structure must be identical to that from Problem 1, i.e. a bank of three correlation receivers, using the same signals $\phi_i(t)$ from part (b). Obviously the decision rule will have to be modified to decode one of eight possible symbols from the three receiver outputs.
- The signals $y_i(t)$ must have the same duration (T) and peak amplitude ($\pm A$) as the signals $s_i(t)$ from Problem 1. (To be clear, some of the $y_i(t)$ will have peak positive amplitude A, and others will have peak negative amplitude $-A$, but it is not necessary that they achieve both $\pm A$.)
- The signals $y_i(t)$ must be chosen so as to optimize the performance of the system, i.e. minimize the symbol error rate.

(a) Determine the set of signals $y_i(t)$ that should be used for this problem, and sketch them vs. t.

(b) Use MATLAB to plot the signal constellation, using the same values for R and A from Problem 1(f).

(c) Determine the decision rule to choose which signal was transmitted from the outputs $\{r_1, r_2, r_3\}$ from the correlation receivers, so that the symbol error rate is minimized.

(d) Determine the probability of symbol error for this system.

3. In this problem, you will approximate the PSD for random polar NRZ data through simulation. (Read Couch, Section 6-2, with particular emphasis on Subsection "Measurement of PSD, Numerical Computation of the PSD.")

(a) Use the `rand` function in MATLAB to generate a sequence of 32 random bits. You will have to round the output of `rand` to 0 or 1.

(b) Use the random data to generate a polar NRZ signal in MATLAB, using rectangular pulses. Create 16 samples of the signal for each bit, using a value of +5 to represent a 1, and -5 to represent a 0.

The total number of samples in your signal vector should be $32 \times 16 = 512$. If your signal includes "tails" of zero values prior to the first bit, and after the last bit, then you should delete those samples from your signal vector before proceeding.

(c) Use MATLAB to compute the spectrum of your signal using the FFT, with the following parameters: $N = 512$, $R = 100$ bits per second, $f_s = 16R$, $\Delta t = 1/f_s$, $T = N\Delta t$.

(d) Let $s(t)$ be the signal you generated in part (b), and let $S(f)$ be the spectrum you computed in part (c). Compute the approximate the PSD of $s(t)$ using

$$P_s(f) = \frac{|S(f)|^2}{T},$$

where T is the duration of $s(t)$ in seconds. Plot the PSD versus frequency in Hertz. Also plot on the same pair of axes the theoretical PSD given by equation (3-41).

(e) The method you have used to approximate the PSD is really only valid if the results are averaged over many random signals. Modify your MATLAB code so that it computes an average PSD for 10 signals. You should create a loop, such that each time through the loop your code generates new data, creates a new signal $s(t)$, computes a new spectrum $S(f)$, and updates the average PSD. Plot the average PSD for 10 signals versus f, and plot equation (3-41) on the same axes.

(f) Repeat part (e) with 100 signals.

(g) Does the agreement between theory and simulation increase as more signals are generated? Is the agreement equivalent for all frequencies? Why or why not?

(h) For the PSD using 100 signals, prepare a plot showing both the theoretical and simulation results for frequencies from 400 to 800 Hertz. Comment on the source of any disagreement here.