Problem set 4. Graphics User Interface

Handed out: Friday, November 8, 2024.

Due: 23:55pm, Friday, November 22, 2024.

You must hand in two files:

- One file, named your_name_E4.pdf, with the solution to the exercise in this set, following the template available in the virtual campus. You must explain how you solved it, the necessary MATLAB code, the results obtained (run transcripts or plots), and some final comments (if any).
- The second file, named your_name_E4.zip, must contain all the MATLAB code you used to solve the exercise, organized in one or more text (.m) files and (.fig) files to allow the teacher to check your solution.

Please, make sure that the code in the pdf file is exactly the same as in the zip file. Remember that all the MATLAB code is supposed to be entirely yours. Otherwise, you must clearly specify which parts are not, and properly attribute them to the source (names of collaborators, links to webpages, book references, ...). Read the Course Information document for more details on this subject.

Binary polar PAM signal generation and matched filter decoding.

The binary polar PAM modulation received in noise can be expressed as:

$$r(t)=\sum_{k=0}^{N-1} b[k]p(t-kT)+w(t),$$

where $r(t)$ is the received signal expressed as a function of the continuous variable t representing time, $b[k]; k=0:N-1$ is the binary random symbol sequence with equally likely values $\pm A$ and $p(t)$ is the rectangular modulation pulse[1]. The noise is assumed to be zero mean white and Gaussian with variance equal to σ_w^2.

The received signal is sampled in time using a sampling period $T_s = T/Q$, where T is the symbol period Q is an integer that determines the number of samples taken per symbol period. The sampled signal is represented in $r[n]=r(nT_s)$ where n is an integer number.

$$r[n]=r(nT_s)=\sum_{k=0}^{N-1} b[k]p(nT_s-kT)+w(nT_s)=\sum_{k=0}^{N-1} b[k]p(nT_s-kQT_s)+w(nT_s)=\sum_{k=0}^{N-1} b[k]p[n-kQ]+w[n]$$

Note that the sampled modulation pulse is now denoted $p[n]$ and is defined as $p[n]=p(nT_s)$. The mathematical expression of the pulse given by

$$p_R(t)=\frac{1}{\sqrt{T}}\Pi\left(\frac{t-T/2}{T}\right),$$

[1] Note that functions of continuous variable are defined with parenthesis, like $r(t)$, and functions of integer variable use brackets, like $r[n]$. When using MATLAB, you can only work with vectors and matrices and therefore you need to sample the functions of continuous variable to functions of integer value, and represent them in a vector of samples of proper length.

where $\Pi(t)$=1 for -0.5 ≤ t ≤ 0.5 and $\Pi(t)$=0 otherwise, i.e., a normalized rectangular pulse centered at 0 of duration 1, so that rectangular pulses are only defined between t=0 and t=T.

a) **Build** a MATLAB function that generates the samples of the received PAM signal in noise having as input parameters:

- The length of the symbol sequence N.
- The symbol period T (in seconds).
- The oversampling value Q (must be integer).
- The amplitude A.
- The signal to noise ratio, SNR (in dB).

In order to determine the SNR, defined as the ratio between the signal and noise powers, you should take into account the expression of the variance of the received signal:

$$E[r[n]^2] = E\left[\left(\sum_{k=0}^{N-1} b[k]p[n-kQ] + w[n]\right)\left(\sum_{l=0}^{N-1} b[l]p[n-lQ] + w[n]\right)\right] =$$
$$\sum_{k=0}^{N-1} E\left[b[k]^2\right]p[n-kQ]^2 + E\left[w[n]^2\right] = A^2\sum_{k=0}^{N-1} p[n-kQ]^2 + \sigma_w^2$$

Which, for the given rectangular pulse, gives:

$$E[r[n]^2] = \frac{A^2}{T} + \sigma_w^2$$

In the previous equation, the first term of the sum corresponds to the signal power and the second term to the noise power. The SNR in dB is defined as 10 times the base-10 logarithm of the signal power divided by the noise power.

b) **Test** this function plotting the generated signal $r[n]$ for different values of the parameters. Identify the samples at multiples of the symbol period.

We now consider the receiver design consisting on a matched filter, the output of which, can be expressed as:

$$y[n] = r[n] * p[n],$$

where * indicates the convolution operation between vectors r[n] and p[n]. When vectors represent temporal sequences, the convolution operation introduces a delay of $N_p - 1$ samples, being N_p the length of the pulse vector p[n].

c) **Modify** the MATLAB function generated before so that it also provides the matched filter output. Test this function plotting $y[n]$ for different values of the parameters. Identify the samples at multiples of the symbol period.

d) Finally, **develop** a GUI to represent both the PAM signal and the matched filter output. Your GUI design should be **original** and should allow for the manual introduction of values for all the input parameters of the MATLAB function. It should also be initialized with suitable default values that allow visualize the proper operation of the software. In addition The GUI must present the following features:

- The main figure must be proportionally re-sizeable.
- The color of the static text must be the same in figure and in the panels.
- Your name and date should appear in main figure.
- If you click on the “Close” button or select the “Close” option in the menu bar, all the figures (main and animation) must be closed.
- The parameters’ values are changed by means of the sliders and in the edit boxes.