Communication Theory (5ETB0) 2024-2025
Assignment 1

A. Alvarado, J. Lagendijk, C. Álvarez, F. Villenas, M. Aiham*
a.alvarado@tue.nl

Version 1.4

November 26, 2024

Submission Deadline: 3.12.2024 23:59

This is the first assignment for the course Communication Theory (5ETB0). You ar asked to solve (2) mathematical problems and one (1) MATLAB problem. Provide the answers to the two mathematical questions and questions Q3.3 and Q3.4 in a **non-handwritten** PDF report. Please use a regular, single-column document with clear, concise, and consistent answers. Be sure to enumerate your answers (e.g., Q1.1, Q1.2, etc.), meaning do **NOT** make a single block of text or long stories. The program to write the report in does not matter (LaTeX, Word, etc., are all allowed) as long as the document is saved or exported as a PDF file.
The two MATLAB functions, required for Q3.1 and Q3.2 have to be placed into a ZIP folder named `xxxxxxx_Assignment1.zip`, where *xxxxxxx* is replaced by your student number. Do NOT rename the two provided functions, `Pe16QAM.m` and `mlDecision16QAM.m`. Naming the ZIP file or the MATLAB files incorrectly could result in not receiving the points for Question 3.

Upload the report as a PDF file to CANVAS in Assignment 1. You are expected to insert your input in the provided MATLAB functions and submit the completed scripts with your report in a ZIP file.

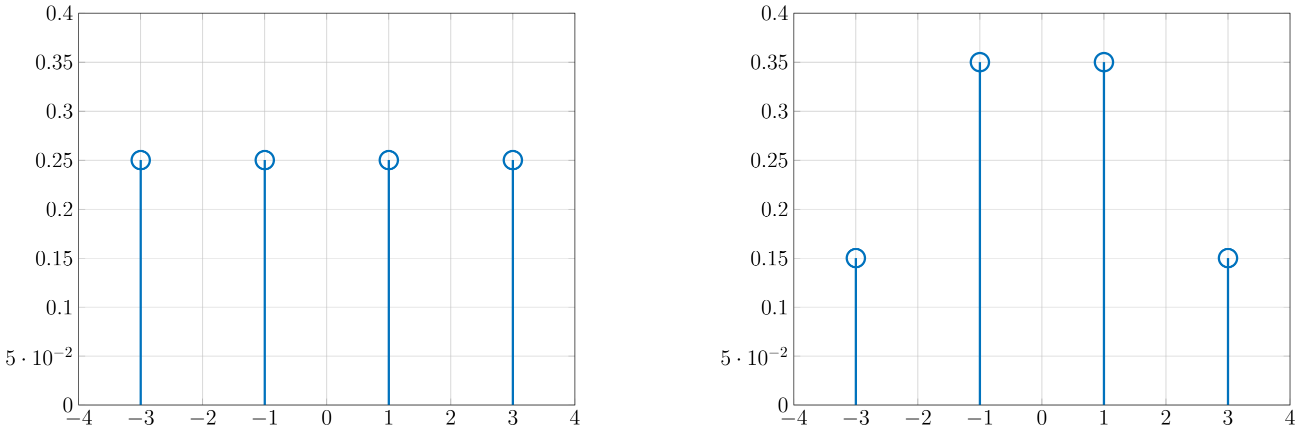
To ask questions concerning the assignments, contact the TAs. Group discussions are encouraged. However, providing the same answers in the report is not allowed. Plagiarism software will be used to check the MATLAB codes and reports.

*Information and Communication Theory Lab, Signal Processing Systems Group, Department of Electrical Engineering, Eindhoven University of Technology, The Netherlands, www.sps.tue.nl/ictlab/

1 Mathematical Problems

Problem 1: 4-PAM Modulation

Consider a DICO channel with additive Gaussian noise (AGN) with noise variance σ^2. The transmitter and receiver use one-dimensional symbols. The set of possible transmitted symbols is $s_m \in \{-3, -1, 1, 3\}$. For the probability distribution of the symbols two options are considered, equiprobable and Gaussian. The two distributions and the corresponding probabilities are shown in Fig. 1.



(a) Equiprobable distribution

(b) Gaussian distribution

Figure 1: The two probability distributions

Q1.1. How many bits can each symbol represent?

Q1.2. Using the equiprobable symbol distribution, which of the decision rules result(s) in minimizing the error probability. Calculate the decision thresholds of an optimum receiver.

Q1.3. Given the Gaussian symbol distribution, which of the decision rules result(s) in minimizing the error probability. Calculate the decision thresholds of an optimimum receiver.

Q1.4. Given noise variance $\sigma^2 = 1$, calculate the average symbol error probability of both distributions for the chosen decision rule.

Q1.5. Calculate the average symbol energy for both distributions.

Problem 2: 8-PSK Modulation and Repetition Coding

Consider a wireless communication channel using 8-PSK modulation. The symbols are defined by a magnitude $E_i = 1$ and an angle $\theta_i = i\pi/4$ for $i = 1, ..., 8$. Each symbol in the system represents a unique vector of 3 bits. All bits and symbols are assumed equiprobable.

Q2.1. Draw the 8-PSK constellation and the decision regions I_1 through I_8 for the optimum receiver. Group these decision regions into groups with the same symbol error probabilities.

Two methods are proposed to map the bits to the symbols, binary mapping and gray mapping. Binary mapping maps each bit according to a binary sequence, i.e. {000, 001, 010, ..., 111}. Gray mapping maps the symbols in such a way that any two adjacent symbols differ at most one bit.

Q2.2. For the two proposed mappings draw the constellation and label all the symbols with the corresponding bit vector.

Q2.3. Which of the two mappings do you expect to have the lowest bit error probability? Explain your answer.

With the aim of reducing the bit error probability (3,1) repetition coding is introduced. This coding schedule repeats each bit three times. There exist then two possible codewords $0 \rightarrow 000$ and $1 \rightarrow 111$.

Q2.4. Given a received sequence of three bits, consisting of a single codeword with potential errors, explain the optimal decision rule for the decoder.

Q2.5. Given a channel with a bit error probability 0.25. Assume that the probablity distribution of each bit is i.i.d., what is the bit error probability after decoding using the optimal decision rule proposed in Q2.4.

2 MATLAB Problem

The ZIP folder `5ETB0_2024_2025_Assignment1_MATLAB_Files.zip` contains 3 files, the main simulation file `SimulatePe16QAM.m`, and two function files `Pe16QAM.m` and `mlDecision16QAM.m`. When handing in your report also include a ZIP folder named `xxxxxxx_Assignment1.zip`, where *xxxxxxx* is replaced by your student number. This ZIP folder must contain your implementation of the two functions, `Pe16QAM.m` and `mlDecision16QAM.m`. Naming the ZIP file or the MATLAB files incorrectly could result in not receiving the points for the relevant question.

Problem 3: Error Probability of 16-QAM

In Chapter 4.8 of the course reader and in Module 4.3, it is shown that the maximum likelihood (ML) decision rule for additive Gaussian noise (AGN) vector channels is the minimum squared Euclidean distance decision rule:

$$\hat{m}_{\text{ML}}(\underline{r}) \triangleq \underset{m\in\mathcal{M}}{\operatorname{argmin}} \left\{ \|\underline{r} - \underline{s}_m\|^2 \right\}. \tag{1}$$

Consider a two-dimensional AGN channel where the transmitted messages are equally likely. The transmitted signal vector corresponding to the message m is $\underline{s}_m$. The received signal is $\underline{r} = \underline{s}_m + \underline{n}$, where $\underline{n}$ represents the AGN with variance σ^2 in both dimensions.
Consider the case where the signal vectors are

$$\underline{s}_m = (x,\ y)\,, \tag{2}$$

where $x \in \sqrt{\frac{P}{10}}\{-3,-1,1,3\}$ and $y \in \sqrt{\frac{P}{10}}\{-3,-1,1,3\}$, being P the average symbol energy. There exist a total of 16 symbols, i.e. $m = 1,\dots,16$. The resulting signal constellation is a 16-QAM constellation.

Q3.1. **(MATLAB Function) Analytical Calculation of 16-QAM Symbol Error Probability:**
When demodulating a 16-QAM symbol using the maximum likelihood decision rule a symbol error occurs when the noise pushes the received symbol outside of the decision region of the transmitted symbol. As 16-QAM is a 2-dimensional format, there are 2 noise components, n_x and n_y. The probability that one of the noise components pushes a symbol over the decision boundary, in a single direction, is given by

$$\Pr\{n_x \geq d/2\} = \Pr\{n_y \geq d/2\} = Q\left(\sqrt{\frac{P}{10\sigma_x^2}}\right) = Q\left(\sqrt{\frac{2P}{10\sigma^2}}\right). \tag{3}$$

Note: Since $\sigma_x^2 = \sigma_y^2$, the total variance σ^2 can be expressed as $\sigma^2 = 2\sigma_x^2$.

Implement the MATLAB function `Pe = Pe16QAM(P,sigma2)`, which analytically calculates the union bound average symbol error probability. The function has the following **input** parameters: `P` is the average power of the signal vectors, and `sigma2` is the total noise variance σ^2. The **output** parameter `Pe` is the probability of error given in (3).

Hint: use the MATLAB function `qfunc(.)` to calculate the Q-function $Q(x) = \frac{1}{\sqrt{2\pi}}\int_x^\infty \exp\left(-\frac{u^2}{2}\right) du$.

Q3.2. **(MATLAB Function) ML Detector for 16-QAM under AGN:**
Implement the MATLAB function `s_hat = mlDecision16QAM(r,P)` which computes the ML decision based on (1). The function has the following **input** parameters: `r` is a single received vector $\underline{r}$ (column vector), and `P` is the power of the signal vectors. The **output** parameter `s_hat` is the estimated transmitted signal $\underline{s}_m$ given the received vector.

An example of the correct function call with all the parameters can be seen in the `SimulatePe16QAM.m` script.

Q3.3. **Visualization of the Results:**
Use the MATLAB script `SimulatePe16QAM.m` to test your functions. Do not forget to include the functions you prepared in the same directory as the provided script. This script estimates by simulations the probability of error and then plots the theoretical probability of error and the simulated one as a function of power on the same figure. In `SimulatePe16QAM.m`, the probabilities of error are calculated for different power levels while using a fixed noise variance $\sigma^2 = 1$, i.e., the signal-to-noise ratio (SNR) can be defined as $\text{SNR} = \frac{P}{\sigma^2} = P$. In your report, you should add two figures generated by `SimulatePe16QAM.m`: The first figure is the resulting plot for $N = 250$, where N is the number of simulated symbols, and the second figure is the resulting plot for $N = 50000$. Note that after each simulation, the time it takes to conduct the simulation is printed on the command window. Take note of these values as you simulate for different N.

Q3.4. **Analysis of the Results:** Briefly answer the following questions in your report:

- Comparing the plots for $N = 250$ and $N = 50000$, which one of these scenarios yields more accurate simulation results? What is the tradeoff between these selections?
- The nearest neighbor approximation used to derive (3) assumes we can mistake an actual transmitted symbol with only one of its neighbors, i.e., it does not include the errors associated with points that are farther away. For which SNR regimes would you expect this approximation to be more valid? Now, compare the plots in the figure where $N = 50000$ is used. For which SNR regime is this approximation more valid? Compare your expectations and the results you obtained.