ETH zürich

Exercise 5: SNR, Detection, and Error Rates

© 2024
Christoph Studer (studer@ethz.ch)
Stefan M. Moser (moser@isi.ee.ethz.ch)

- **Exercise: Thursday, 7 November 2024**
- This exercise will be assisted in CAB G11 from 8:15 h to 10:00 h
- Problem 4 requires a computer with MATLAB (or Octave) installed
- **Due date:** Monday, 18 November 2024, 08:00 h
- **Submission instructions:** Upload your solutions (answers and MATLAB code) to Moodle as a *single* zip-file with a file name `⟨last-name⟩-⟨first-name⟩-e5.zip`. Your homework should consist of a single pdf file for the written answers and one MATLAB script per problem part. The MATLAB file must be named according to the syntax `E5_P4_3.m` (for "Exercise 5 Problem 4 Part 3").
- **Rules:** You may collaborate with other students on this exercise, but each student must turn in their own assignment. You must specify the names of your collaborators! Copying answers from previous years' exercises is strictly forbidden.

Problem	Maximum Points	Points Received
1	20	
2	10	
3	20	
4	50	
Total	**100**	

Problem 1: Simple SNR Calculations (20pts)

The goal of this problem is to calculate the SNR for some simple communication channels. Remember that the SNR is defined as

$$SNR = \frac{\text{average received signal energy per complex symbol time}}{\text{average received noise energy per complex symbol time}}. \tag{1}$$

Part 1 (10pts). Assume the following simple AWGN channel $y = x + n$, where $x \in \{0, 1\}$ is the transmit signal with equally likely symbols and $n \sim \mathcal{N}(0, \sigma^2)$ models AWGN. Assume that the noise is independent of the transmit signal. Compute the SNR for this channel.

Part 2 (10pts). Assume the following channel $y[k] = -0.6x + n[k]$ with two time slots $k \in \{1, 2\}$. We assume that $x \in \{-3, -1, +1, +3\}$ and $n[k]$ IID $\sim \mathcal{N}(0, \sigma^2)$ and that the noise and the signal are independent. Compute the SNR.

Problem 2: Conversion of AWGN into Binary Symmetric Channel (10pts)

Consider an AWGN channel modeled as $y = s + n$, where $s \in \{-1, +1\}$ are BPSK transmit symbols that are equally likely and $n \sim \mathcal{N}(0, \sigma^2)$ is AWGN. Assume that the receiver consists of a MAP detector that takes the channel output y and generates an estimate $\check{s} \in \{-1, +1\}$. Clearly, the effective AWGN channel from the input s to the MAP detector output $\check{s}$ has binary inputs and binary outputs.

The goal of this problem is to convert this effective AWGN channel into an equivalent binary symmetric channel (BSC) with crossover probability p.

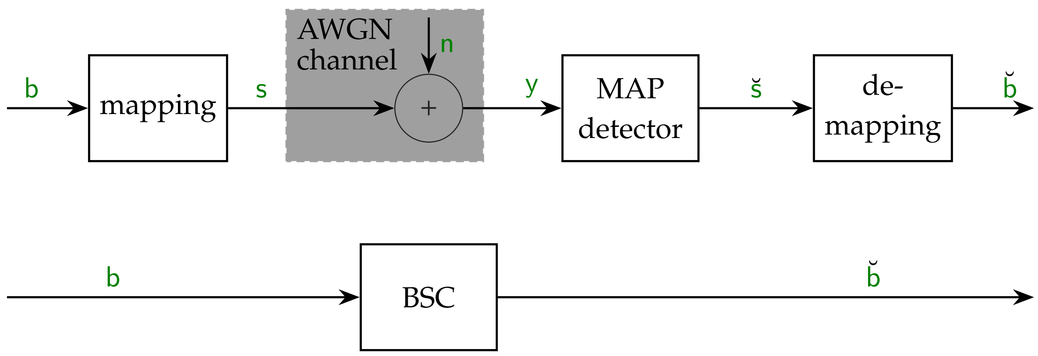

Figure 1: Equivalent BSC of AWGN channel including MAP detector.

In Figure 1, the effective AWGN channel is shown on the left and the equivalent BSC on the right. Assume that the input bit 0 is mapped to the BPSK symbol $+1$ and the bit 1 to the BPSK symbol -1. Remember that a BSC with input B and output $\hat{B}$ has the structure shown in Figure 2.

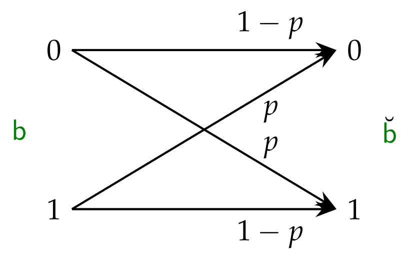

Figure 2: Binary symmetric channel (BSC).

Derive the mathematical relationship between the SNR of the AWGN channel and the crossover probability p of the effective AWGN channel (that includes mapping from bits to BPSK symbols, transmission over the AWGN channel, MAP detection, and mapping from BPSK symbol to bits).

Problem 3: Binary Energy Detection (20pts)

Assume that we have a source that is able to generate two different types of noise: $\mathsf{x}_1 \sim \mathcal{CN}(0, 1)$ and $\mathsf{x}_2 \sim \mathcal{CN}(0, 2)$. The source either generates x_1 or x_2 with probability $\frac{1}{2}$, and you can only observe y which is either x_1 or x_2. One can model the source as follows. First, it generates a Bernoulli random variable $\mathsf{b} \in \{1, 2\}$ where $p_{\mathsf{b}}(1) = p_{\mathsf{b}}(2) = \frac{1}{2}$. Then, the source assigns either x_1 or x_2 to the output y according to the following rule:

$$\mathsf{y} = \begin{cases} \mathsf{x}_1 & \text{if } \mathsf{b} = 1, \\ \mathsf{x}_2 & \text{if } \mathsf{b} = 2. \end{cases} \tag{2}$$

The goal of this problem is to design a detection rule for the Bernoulli random variable b based on the observation y, which decides whether x_1 or x_2 was generated.

Part 1 (10pts). Design the MAP rule for the above system and simplify the decision rule as much as you can. *Hint: Use the conditional PDF $f_{\mathsf{y}|\mathsf{b}}(y|b)$.*

Part 2 (10pts). Compute the error probability P_e of the MAP detector. Provide a numerical value.

Hints: Use the fact that the distribution of $|\mathsf{x}_1|^2$ is exponential with rate parameter $\lambda = 1$ and $|\mathsf{x}_2|^2$ is exponential with rate parameter $\lambda = 2$, and recall that the CDF of a rate-λ exponential RV u is:

$$\mathbb{P}[\mathsf{u} \leq u] = \begin{cases} 1 - e^{-\frac{u}{\lambda}} & \textit{for } u \geq 0, \\ 0 & \textit{for } u < 0. \end{cases} \tag{3}$$

Problem 4: AWGN Channels and Monte–Carlo Simulation (50pts)

Consider the transmission of a binary valued signal from the constellation set $\mathcal{X} = \{x_a, x_b\}$ with $x_a, x_b \in \mathbb{R}$, over an additive white Gaussian noise (AWGN) channel. AWGN channels are modeled as $y = x + n$, where the prior for RV $x \in \mathcal{X}$ is $\mathbb{P}[x = x_a] = p_a$ and $\mathbb{P}[x = x_b] = p_b$ with $p_a + p_b = 1$, and the noise is zero-mean Gaussian with variance σ^2, i.e., $n \sim \mathcal{N}(0, \sigma^2)$.

The goals of this problem are to (i) develop the optimal detector for this system, (ii) derive an analytical expression for the associated error rate, and (iii) use Monte–Carlo techniques to simulate the error rate in MATLAB.

Part 1 (15pts). Derive the detection rule for this AWGN system that minimizes the error probability.

Hint: The situation in this problem is a bit more general than the one shown in class.

Part 2 (15pts). Derive an analytical expression for the error probability for the optimal detector from Part 1.

Hint: The situation in this problem is a bit more general than the one shown in class.

Part 3 (20pts). Simulate the error probability this AWGN system using Monte–Carlo simulations in MATLAB.

In previous exercises, you have already learned the basics of Monte–Carlo simulations. You might remember the fact that $\mathbb{P}[x \neq \check{x}] = \mathbb{E}_x[\mathbb{1}(x \neq \check{x})]$, where $\mathbb{1}(x \neq \check{x})$ is the indicator function that is 1 if a detection error occurs, i.e., if $x \neq \check{x}$, and 0 if the detector generates the correct transmit signal, i.e., if $x = \check{x}$.

We are interested in approximating the error rate with the sample mean

$$\mathbb{P}[x \neq \check{x}] = \mathbb{E}_x[\mathbb{1}(x \neq \check{x})] \approx \frac{1}{T} \sum_{t=1}^{T} \mathbb{1}(x_t \neq \check{x}_t), \tag{4}$$

where $x_1, x_2, \ldots, x_T$ are samples generated independently from the distribution of RV x. Specifically, the following steps must be carried out for every Monte–Carlo trial $t = 1, \ldots, T$:

1. Generate a transmit symbol x_t from the set $\{x_a, x_b\}$ with priors p_a and p_b, respectively.
2. Generate a zero-mean Gaussian noise sample n_t with variance σ^2.
3. Compute the received signal $y_t = x_t + n_t$ as defined by the AWGN model.
4. Use the optimal MAP detector from Part 1 to generate an estimate $\check{x}_t$ from y_t.
5. Evaluate $\mathbb{1}(x_t \neq \check{x}_t)$.

After repeating the above steps for T times, one can approximate $\mathbb{P}[\mathsf{x} \neq \check{\mathsf{x}}]$ using the right-hand side expression in (4). For this part, assume $x_{\mathrm{a}} = -1$ and $x_{\mathrm{b}} = +1$ and uniform priors $p_{\mathrm{a}} = p_{\mathrm{b}} = 0.5$.

Write a MATLAB script named "`E5_P4_3.m`" that implements the above-described Monte–Carlo simulation. Simulate the error probability for different signal-to-noise ratio (SNR) values, i.e., for $SNR_{\mathrm{dB}} \in \{-10, -9, \ldots, 19, 20\}$ decibel; remember that the SNR for this problem is

$$SNR = \frac{\mathbb{E}_{\mathsf{x}}\left[\mathsf{x}^2\right]}{\sigma^2} \tag{5}$$

and

$$SNR_{\mathrm{dB}} = 10 \log_{10}(SNR). \tag{6}$$

Perform $T = 10,000$ Monte–Carlo trials for each value of SNR_{dB}. Plot the resulting error probability function $P_{\mathrm{e}}(SNR_{\mathrm{dB}})$ in MATLAB as a doubly-logarithmic plot (the SNR values are already in decibels). The SNR in decibels range should be from -10 to $+25$ and the error rate range should be from 10^{-6} to 10^{0}. Include the analytical error rate from Part 2 in your plot and compare them—do the two curves match?