Binary Phase-Shift Keying (BPSK) modulation

Laboratory 3, DEPI

Objective

Simulate a BPSK modulation system and its decoding performance in white gaussian noise.

Theoretical aspects

Binary Phase-Shift Keying (PSK) modulation is a binary encoding procedure defined as follows:

- for a logical bit 0, send the signal $s_0(t) = A\sin(2\pi f t)$
- for a logical bit 1, send the signal $s_1(t) = A\sin(2\pi f t + \pi) = -A\sin(2\pi f t)$

The difference is in the sign (or phase π) of the signal.

The duration of a bit signal is a multiple of the sine period, $T_{bit} = \frac{1}{f} \cdot k, \quad k \in \mathbb{N}$

The receiver recovers the data by multiplying with $\sin(2\pi f t)$ and integrating the result

$$\pm \int_0^{T_{bit}} A\sin(2\pi f t)\sin(2\pi f t) dt = \pm A \int_0^{T_{bit}} \frac{1 - \cos(2\pi(2f)t)}{2} dt = \pm \frac{A T_{bit}}{2}$$

The sign of the result matches the sign of the original signal:

- If the result is positive, the original bit is 0 (signal amplitude was A)
- If the result is negative, the original bit is 1 (signal amplitude was $-A$)

When the received signal has noise, the values might vary. There can be 4 outcomes:

- correct rejection: original bit is 0, detection is 0
- false alarm: original bit is 0, detection is 1

- miss: original bit is 1, detection is 0
- hit (correct detection): original bit is 1, detection is 1

In general, the result can be compared to a threshold T which might not necessary be 0, but closer to one value or the other. If the distribution of the noise is known, a precise value for T can be found with a decision criterion.

The performance of the detection scheme can be summarized in a Receiver Operating Characteristic (ROC) plot. The Receiver Operating Characteristic (ROC) curve is the plot of P(hit) against P(false alarm).

Exercises

1. Simulate the BPSK sender
   - Generate a vector **`data`** of 1000 values 0 or 1, with equal probability (hint: use `rand()` and compare to 0.5).
   - Generate a vector **`signal`** of 100000 values as follows: for each bit in **`data`**, put a 100-long sine $\pm A \sin(2\pi f n)$ in **`signal`**. Use $A = 1$, $f = 1/100$. 0 corresponds to $+A$, 1 to $-A$.
   - Plot the resulting signal.
2. Simulate a noisy channel
   - Generate a vector of white gaussian noise with distribution $\mathcal{N}(0, \sigma^2)$, the same length as `signal`, and $\sigma^2 = A/10$.
   - Add the noise to the signal, store result as **`signalplusnoise`**.
3. Simulate the BPSK receiver
   - For each 100-long piece from **`signalplusnoise`**, multiply the piece element-wise with $\sin(2\pi f n)$, and sum it. Put the sum results, for each piece, into a 1000-long vector **`integrals`**.
   - Decide the bit vaues by thresholding **`integrals`** with a threshold T. Use the value $T = 0$. The result will be a binary vector **`decoded`**.
   - Compare **`decoded`** with the original **`data`** and compute the probability of *hit* and the probability of *false alarm*.
4. Draw the ROC.
   - Wrap all the above into a function `[phit pfa] = BPSKsim(T)` and call it for different values of T, going from $-50A$ to $50A$.
   - Store the results for each case, and at the end plot the graph P(hit) as a function of P(fa).

Final questions

1. When should a value $T \neq 0$ be used?