

S-72.341 Coding Methods (3 cr) P spring 2004

Home Assignment 3: Viterbi Decoder (optional)

This assignment was developed by Tk.L. Jari Salo (jari.salo@hut.fi)

1 Task

Your task is to implement, in MATLAB or C, a hard-decision Viterbi decoder for a rate- R convolutional code given in Table 1. A received signal sample is $z_r = z + n$, where the transmitted BPSK symbol $z \in \{-1, +1\}$ and the noise sample n is from zero-mean Gaussian distribution with variance $\sigma_n^2 = \frac{1}{2RE_b/N_0}$. E_b/N_0 is the signal-to-noise ratio per information bit. The receiver makes hard decisions $\{0, 1\}$ of received samples before passing them to the Viterbi decoder.

Table 1: Code polynomials in octal presentation. μ is the sum of the first five digits of your student ID modulo 10. Example: student ID =12345A, $\mu = 5$

μ	code polynomials
0 or 1	5 7
2 or 3	15 17
4 or 5	23 35
6 or 7	13 15 17
8 or 9	25 33 37

2 Grading

The assignment will be graded on the scale of 0-3 points. These points will be added to your exam score for all exams until spring 2005. Criteria: 0=decoder does not work and/or report is deficient, 1=decoder works correctly, 2=decoder works correctly and efficiently + nice report, 3= previous + implementation is particularly clever and of general usage.

Once you have submitted your report and computer code, you cannot improve it any more in hope of a better grade.

3 How to pass the assignment?

The assignment is to be completed individually within the bounds of the following honor code: You are allowed to discuss and solve implementation problems with other students,

but in your report you should credit your collaborators and their contributions. You must write your own computer code and report.

In order to pass the assignment you must:

- Return your computer code by e-mail to *nikolai.nefedov@hut.fi* no later than on 28.4.2004. All computer code required to run your program, including any auxiliary functions or scripts (including `chain.m`) is to be included. In the subject field of your e-mail write "S-72.341 assignment/12345A", where 12345A is to be replaced with your student number.
- Return your written report no later than on 28.4.2004 to the box below the course bulletin board in the E wing, 2nd floor, of the ECE building.

The deadline is definite (no exceptions).

4 Requirements for the written report

Write a maximum two page long report of your results. In the report you must include: i) your name, student id number, and email address; ii) your code parameters; iii) brief comments on what you considered the most difficult part in the assignment; iv) a graph of your simulation results compared to theoretical values; v) all computer code in appendix. In the graph you must include the following:

- a simulated *BER* curve of your code, with values down to $BER = 10^{-4}$ or less
- theoretical *BER* curve for uncoded BPSK. What is the coding gain at $BER = 10^{-4}$?
- Shannon capacity limit for any rate- R code (code rate R depends on your code). What is the gap to capacity at $BER = 10^{-4}$?
- capacity of a binary symmetric channel (BSC) for rate- R . What is the gap to BSC capacity at $BER = 10^{-4}$?

Figure 1 shows how the graph looks like for a hypothetical rate-1/2 coding scheme.

5 Comments and hints

- You may take advantage of any and all MATLAB functions - except `vitdec` - to help you to program your decoder. Additionally, you are given the script file `chain.m` that implements all other parts of the simulation chain, apart from the Viterbi decoder. The script can be downloaded from the course home page.

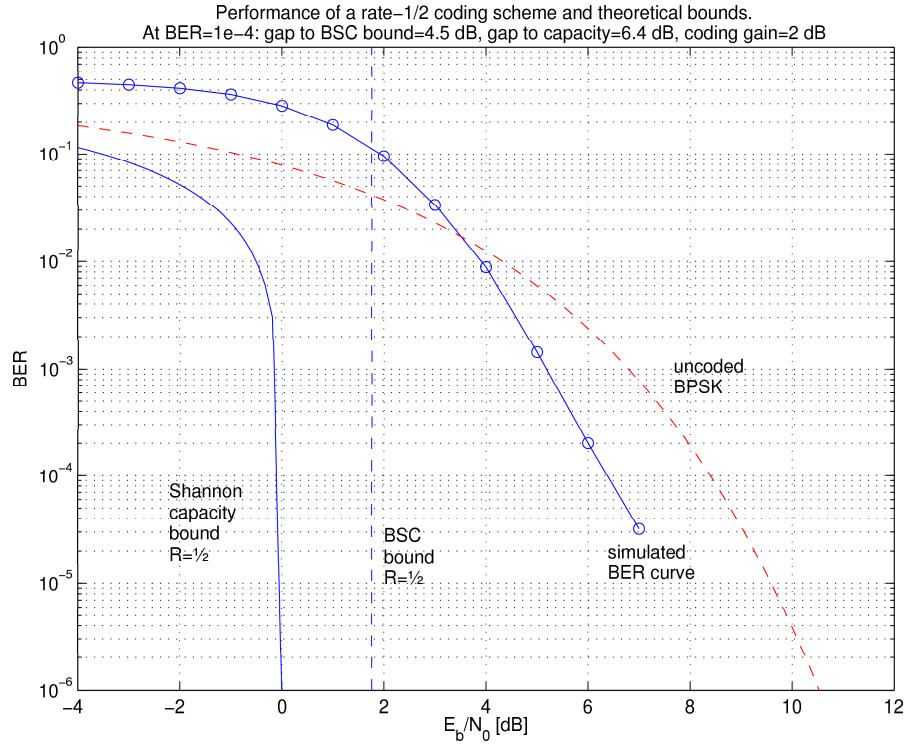


Figure 1: Performance of a rate-1/2 coding scheme and theoretical bounds.

- If you opt to implement your decoder in C you can still utilize `chain.m`, but you will have to create a MATLAB-C gateway routine to call your C function from MATLAB.
- For statistically reliable BER estimates a sufficiently large number of information bits should be simulated per each E_b/N_0 value. As a rule of thumb you should decode a minimum of $100/BER$ information bits per BER point. For example, to simulate $BER = 10^{-4}$ you should decode at least 10^6 bits.
- Shannon capacity limit can be computed from $R = \frac{\log_2(1+2RE_b/N_0)}{2[1-H(BER)]}$, where $H(x) = -x \log_2(x) - (1-x) \log_2(1-x)$. BSC bound can be computed from $R = 1 - H(BER)$, where $BER = Q(\sqrt{\frac{2RE_b}{N_0}})$ is the cross-over probability of the BSC assuming BPSK modulation and a rate- R coding scheme.