Applications of Information Theory (5LSF0) 2023-2024

G. Liga*

Assignment #3
Module 4: Convolutional Codes and BCJR algorithm

In this assignment, you will simulate the performance of convolutional codes using a soft-decision decoder based on the BCJR algorithm.

1 Description

Let us assume coded transmission of binary data over a BI-AWGN channel defined by a signal-to-noise (SNR) ratio E_s/N_0. The adopted code is a convolutional code defined by the defined by the following generator matrix in polynomial form:

$$G(D) = [1 + D^2, 1 + D + D^2]. \tag{1}$$

In the matrix (1), each entry (i, j) is the generator polynomial representing the relationship between input i and output j of the convolutional encoder. On the receiving side, we assume we decode the received sequences using a BCJR decoder followed by hard decisions on the bits.

2 Questions

Answer the following questions:

1. What class of convolutional codes does the one described in (1) belong to? Feedforward systematic, recursive non-systematic, else?
2. What is the code rate?
3. What is the encoder memory?
4. How many states do you need to fully describe the encoder?

3 Simulation Tasks

Execute the following simulation tasks:

1. Estimate via Monte-Carlo simulations the coded bit error rate (BER) for the code in (1) using a BCJR decoder. Plot the BER as a function of E_b/N_0.
2. Estimate the net coding gain (NCG) of the code (1) at BER=10^{-6}.

   Instructions: For the implementation of the convolutional encoder use the MATLAB function *convenc*. Pay particular attention to the specification of the generator polynomials. You are free to choose whether to terminate (and how) the encoded sequences or not. Implement the BCJR algorithm in the log-domain. Carefully consider the initialisation of the BCJR quantities based on your encoding choices. Flexibly choose the buffer length of the BCJR decoder provided that the coded performance is not affected. As usual, make sure the BER vs E_b/N_0 curve is smooth.
3. Repeat tasks 1 and 2 for the code

$$G(D) = [1 + D^2 + D^3, 1 + D + D^2 + D^3]. \tag{2}$$

   Does the code in (2) perform better or worse than the previous one? What about the NCG at BER=10^{-6}? Can you explain this behaviour?

*Information and Communication Theory Lab, Signal Processing Systems Group, Department of Electrical Engineering, Eindhoven University of Technology, The Netherlands, http://www.tue.nl/ictlab, contact email: g.liga@tue.nl