Applications of Information Theory (5LSF0) 2023

G. Liga, K. Wu*

May 23, 2023

Assignment #4
Module 4: LDPC Codes

1. Consider the decoding of the jth VN of an LDPC code. Two LLRs are calculated at a given iteration: L_j^{total} and $L_{j\to i}$. Answer the following questions:
   (a) Which of these LLRs should be used for making a final decision about the bit c_j. Why?
   (b) What rule should be applied to the LLR to obtain an estimate of the transmitted bit c_j?
   (c) How can the estimated bits be used for stopping the decoder?

2. An LDPC decoder needs to be initialized using the channel LLRs, which represent the a posteriori probabilities of the code bits. For a binary erasure channel with channel observations $y_j \in \{0, 1, e\}$, where e represents an erasure and $j = 0, 1, \dots, n-1$, these a posteriori probabilities are:[1]

$$\Pr(v_j = b|y_j) = \begin{cases} 1, & \text{if } y_j = b \\ 0, & \text{if } y_j = 1-b \\ 1/2, & \text{if } y_j = e \end{cases}, \tag{1}$$

   where $b \in \{0, 1\}$. In this case:
   (a) Give a step-by-step derivation[2] of the probabilities in (1).
   (b) Explain why the LLRs used for the initialization of the LDPC decoder should be $\pm\infty$ or 0.

3. Consider the DVB-S2 LDPC code with rate $R_c = 5/6$, whose parity check matrix is provided by the Matlab function "dvbs2ldpc".
   (a) Is this code regular or irregular? Give the degree distribution polynomials. Write a Matlab script that computes the λ_d and ρ_d values.
   (b) Simulate the performance of this code over the bi-AWGN channel considering two algorithms: (i) The BP algorithm (use Matlab's implementation), and (ii) the bit-flipping (BF) algorithm (see below) based on hard-decisions made on the received symbols. For the BF algorithm, set the maximum number of iterations to 100, and repeat your simulations for different values of δ.

   Bit-flipping (BF) algorithm is a low-complexity decoding algorithm proposed by Gallager in his original paper [1] in which LDPC codes were introduced. It is based on hard-decisions $\widehat{\boldsymbol{v}}$ made on the received symbols $\boldsymbol{y}$. First, for each hard-decision $\widehat{v}_i$ for $i = 1, 2, \dots, n$, the number of unsatisfied parity check equations w_i that they are a part of is found. Second, the bits $\widehat{y}_i$ with $w_i > \delta$ are *flipped*, i.e., $0 \to 1$ or $1 \to 0$. Then this procedure is repeated iteratively until either all parity checks are satisfied or a pre-defined maximum number of iterations is reached.

[1] R.G. Gallager, "Low-density parity-check codes," *IRE Trans. Inf. Theory,* vol. 8, no. 1, pp. 21–28, Jan. 1962.

*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, k.wu}@tue.nl

[1] Hint #1: See page 219 of Lin and Ryan "Channel Codes: Classical and Modern".

[2] Hint #2: Draw a channel and identify the transition probabilities and use Bayes' rule for $b = 0$ and $b = 1$.