TSKS12 Modern Channel Coding, Inference and Learning

Laboration 2:

Performance Analysis of Turbo Codes over AWGN channels

Objective: The aim of this exercise is to investigate the bit error rate performance of turbo codes over AWGN channel when BPSK modulation is used.

Background: The turbo codes are near Shannon capacity error correcting codes. They use a parallel concatenated encoding scheme in which two recursive systematic convolutional (RSC) encoders are connected in parallel and are separated by an interleaver, i.e., they consist of two or more component encoders separated by an interleaver so that each encoder uses an interleaved version of the same information sequence. The choice of the interleaver affects the code design as it determines the distance properties of the code. A good linear code is one that has mostly high weight codewords since they are then more distinguishable and the decoder can estimate the bits more reliably. So, the job of the interleaver is to scramble bits before being input to the second encoder. This makes the output of one encoder different from the other encoder. Thus, even if one of the encoders occasionally produces a low weight sequence, the probability of both the encoders producing a low weight output is extremely small.

Turbo Encoder: A turbo encoder comprises of two identical rate $\frac{1}{2}$ RSC encoders concatenated in parallel. The two component RSC encoders are separated by an interleaver. An input information bit sequence is fed to the first RSC encoder which generates a parity sequence as the ouput. The information sequence is also fed to an interleaver and the interleaved information is fed to the second RSC encoder that generates another parity sequence at the output. Thus, the output of the turbo encoder consists of the systematic input data and the parity outputs from the two constituent encoders so that the overall rate becomes one third. A higher rate can be obtained using the same construction by not transmitting some of the bits generated by the encoders. This procedure is known as puncturing.

Modulation and Transmission: The information sequence and the two parity sequences are then modulated and transmitted through the channel. And at each time unit, three values are output from the channel, one for the information bit and two for the parity bits.

Turbo decoder: The two component turbo decoders are based on the BCJR algorithm which provides an estimate of the a posteriori probability of each bit in a codeword based on the received signal. The input to each decoder at each time instant contains three terms, a soft value corresponding to the information bit, one corresponding to the parity bit, and the extrinsic a posteriori L-value from the other decoder which is nothing but the original a priori L-values in the initial iteration of the first decoder. The BCJR algorithm based decoder basically computes and outputs the log likelihood ratio (LLR) $L_{\text{out}}(b) = \ln\left(\Pr[\text{b} = +1|\mathbf{r}]/\Pr[\text{b} = -1|\mathbf{r}]\right)$, where $\mathbf{r}$ denotes the observation fed to the BCJR algorithm. Using the LLR values, the bits can be reliably estimated.

Task I: The first task is to implement a turbo encoder with the following specifications. Let the encoding for each rate $\frac{1}{2}$ RSC encoder be done as follows:

$$G(x) = \begin{bmatrix} 1 & \dfrac{1+x^2}{1+x+x^2} \end{bmatrix}$$

The rate $\frac{1}{2}$ RSC encoder with the above set of generators can be obtained from a non-recursive non-systematic convolutional encoder whose generators are specified as [7,5] in the octal format and whose constraint length is 3. Furthermore, assume that the output from the turbo encoder is modulated using BPSK modulation and sent through an AWGN channel.

Do the following:

1) Obtain the BER vs SNR plots (SNR range: 0 to 6 dB) for the rate $1/3$ Turbo code one for frame length 100, and other for frame length 10000, and for different number of iterations: 1, 2, 5, and 10. Also, plot the performance of uncoded BPSK over AWGN channel in each of the two plots.
2) Perform puncturing to obtain a rate $1/2$ Turbo code and obtain 2 plots as before for frame lengths 100 and 10000, show the BER performance over iterations 1,2 5, and 10 and also for the uncoded case in each of the two plots.

Repeat 1 and 2 to obtain 4 more plots for a constraint length 5 Turbo encoder, in which the encoding is done using the following generator polynomials

$$G(x) = \begin{bmatrix} 1 & \dfrac{1+x+x^2+x^3+x^4}{1+x^2+x^3+x^4} \end{bmatrix}$$

Task II: Implement the BCJR algorithm on the [7,5] RSC encoder and obtain the BER vs SNR plot, run just one iteration.

Some useful Matlab commands and system objects that can be used: *poly2trellis, comm.turboencoder, comm.turbodecoder, comm.BPSKModulator, comm.BPSKDemodulator, comm.AWGNChannel, step.*