Forward Error Correction (FEC) Codes and Their Performance

Introduction:
A communication system without FEC coding is not a good system. In this project, you will simulate two types of FEC codes, evaluate their AWGN performance against the uncoded scheme, and find out what practical systems use those codes.

1) Linear block codes: rate 1/n repetition code; one or two Hamming codes, a rectangular code (or product code), extended Golay code, one or two BCH codes;

2) Convolutional codes: convolutional codes with constraint length K=3, 6, 9; choose block length 32, 128, and 1024; initiate the encoder to the all zero state and pad (K-1) zeros to the end of input bits.

This project will be performed by teams of two members. By the end of the module, your team shall submit a lab report, Matlab programs, and team evaluation form to canvas.

Instructions:

1. Encoder:
   a. Generate independent random bits as the information bits;
   b. Encode the information bits and denote them as vector **c**.
   c. Map the coded bits into BPSK symbols **s**=2***c**-1; this is the BPSK modulation and will guarantee the average bit energy E_c =1.
2. Channel: Add AWGN with specified Eb/No; Note the following:
   a. The conversion between Ec/No and Eb/No is related with the coding rate;
   b. As a guidance, for higher Eb/No, more information bits are needed to simulate the bit error rate (BER) – allow roughly 10 error bits to occur before calculating the BER.
   c. Choose Eb/No such that your BER goes from 0.5 to at least 1e-4.
3. Decoder:
   a. Use hard-decision decoding with a threshold of 0.
   b. Use syndrome decoder for linear block codes;
   c. Use Viterbi decoder for convolutional codes and a trace-back length ~ 3K to 5K;
4. Error performance: save the BERs and their corresponding Eb/No for all coding schemes, plot the BER for each type of the channel coding schemes with the uncoded BER. Compare the results.
5. Report:
   a. discuss error correction capability and coding gain of each code;
   b. why is coding gain possible?
   c. The pros and cons of each code scheme and their use in practical systems.