EE 5501 Prof. N. Jindal
Digital Communication Oct. 23, 2009

Homework 5

Due: Friday, Oct. 30, 5:00 PM

1. In this problem you will implement a Viterbi decoder for the running example $R = 1/2$ binary convolutional code we have been studying in class $(u[k]+u[k-1]+u[k-2], u[k]+u[k-2])$. Matlab code that initiates the Viterbi algorithm, runs an outer loop for different E_b/N_0 values, and that computes the probability of bit error is provided. Your job is to fill in the main portion of the Viterbi decoding algorithm, and to add expressions for the uncoded bit error probability and for the nearest neighbor approximation. Use your code to generate a plot of bit error (actual bit error, uncoded bit error, and nearest neighbor approximation) vs. E_b/N_0. Turn in your completed Matlab code and the numerical plot.
   Note: When you are writing the code, you may want to reduce the number of iterations while you are testing and debugging. When you make the final bit error plot, choose a large enough number of iterations so that the curves are smooth (this will likely require you to run the iteration for some time).

2. We now consider a slightly different $R = 1/2$ code, also with memory 2: $(u[k] + u[k-1] + u[k-2], u[k] + u[k-1])$.

   (a) Draw the trellis diagram for this code.

   (b) By extending the trellis for a suitable number of states, verify that $d_{\text{free}} = 4$ for this code and that there is only a single such simple error event.

   (c) Modify your Matlab code to work for this code, and generate a plot that contains 5 curves: the actual bit error and the nearest neighbor approximation for this code, the actual bit error and the nearest neighbor approximation for the code in the previous question, and the uncoded bit error rate. Explain why this code performs worse than the one in the previous question.

3. We now consider a different $R = 1/2$ code, this time with memory 4:

$$\begin{aligned} y_1[k] &= u[k] + u[k-3] + u[k-4] \\ y_2[k] &= u[k] + u[k-1] + u[k-2] + u[k-4] \end{aligned}$$

   (a) Label the branches of the provided trellis diagram.

   (b) Verify that $d_{\text{free}} = 7$ for this code, and that there are two simple error events with this distance.

   (c) Modify your Matlab code to work for this code, and generate a plot that contains 5 curves: the actual bit error and the nearest neighbor approximation for this code, the actual bit error and the nearest neighbor approximation for the code in

the first question, and the uncoded bit error rate.
Note: To simplify this process, below is a table that tells you the codewords corresponding to all possible starting 4 info bit sequences. This table should be helpful in initializing the Viterbi algorithm.

Info Bits	Codeword
0000	00 00 00 00
0001	00 00 00 11
0010	00 00 11 01
0011	00 00 11 10
0100	00 11 01 01
0101	00 11 01 10
0110	00 11 10 00
0111	00 11 10 11
1000	11 01 01 10
1001	11 01 01 01
1010	11 01 10 11
1011	11 01 10 00
1100	11 10 00 11
1101	11 10 00 00
1110	11 10 11 10
1111	11 10 11 01

Also, when writing the code for Viterbi algorithm, try to identify patterns - if you do so, in some places you will be able to cut and paste different portions of the code and will only have to flip a few signs.