NAME ..................................

Soc. Sec. # ..........................

Remote Location ..........................
(if on campus write campus)

FINAL EXAM

EE568
KUMAR
Sp’00
May 3

- OPEN BOOK exam (students are permitted to bring in textbooks, handwritten notes, lecture notes etc into the exam room).

- calculators are permitted

- 120 min. exam

- notation is as per my lecture notes

- Answer all questions

- Show all intermediate steps clearly; this applies to all questions

- 8 questions, 15 pages, maximum score = 50 points.

- Good luck!
<table>
<thead>
<tr>
<th>SCORE</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Grade on Final Exam</td>
<td></td>
</tr>
<tr>
<td>Grade on Course</td>
<td></td>
</tr>
</tbody>
</table>

1. A rate $\frac{1}{3}$ turbo code is able to provide a bit error probability of $10^{-6}$ at an $E_b/N_0$ of 0.5 dB. If we agree to treat a bit error probability of $10^{-6}$ as providing reliable communication, how many dB away is this from the minimum $E_b/N_0$ required to achieve reliable communication (as given by the channel capacity formula)?

The channel in this case is the discrete additive white Gaussian noise channel modeled by

$$r_i = s_i + n_i, \quad i \geq 0$$

where $r_i$ is the $i$th received signal, $s_i$ the $i$th transmitted signal and $\{n_i\}$ is a sequence of independent, identically distributed Gaussian random variables (the noise) having zero mean and variance $N_0/2$.

4 points
2. What is the nominal coding gain of a rate $\frac{1}{3}$ turbo code having minimum free distance $d_{\text{free}} = 8$? (Communication is over the same discrete additive white Gaussian noise channel described in the previous question).

2 points
3. Consider the convolutional code encoded by polynomial generator matrix

\[ G(D) = \begin{bmatrix} 1 + D & 1 + D^2 + D^3 & D^2 \\ 1 & 1 + D & 1 \end{bmatrix}. \]

(a) What is the external degree of the convolutional encoder?  
2 points

(b) What is the internal degree of the convolutional encoder?  
2 points

(c) What can you say concerning the minimum number of delay elements required to implement the encoder?  
2 points
4. Does the convolutional encoder having polynomial generator matrix

\[ G(D) = \begin{bmatrix} 1 + D & 1 + D^2 + D^3 & D^2 \\ 1 & 1 + D & D \end{bmatrix} \]

have catastrophic error propagation? 

4 points
5. Consider the convolutional code having polynomial generator matrix

\[ G(D) = \begin{bmatrix} D & 1 + D \end{bmatrix}. \]

(a) Draw the state and trellis diagrams (out to the 4th node level, starting from node level 0) of this code. As in class, use dotted lines to represent a message symbol that is 1 and a solid line otherwise. Label each branch of the trellis with the corresponding code symbols. 4 points

(b) Determine the generating function \( A_F(L, I, D) \) of the code (notation as in class). 4 points

(c) What is the free distance \( d_{\text{free}} \) of the convolutional code? Explain how you determined \( d_{\text{free}} \). 2 points
Extra workspace
6. Determine an upper bound to the bit error probability $P_{be}$ of the rate $\frac{1}{2}$ convolutional code having generating function

$$A_F(L = 1, I, D) = \frac{I^2D^6 + I^3D^5 - I^4D^6}{1 - (2ID - I^2D^2 + D^2)}$$

The channel in this instance is a binary symmetric channel having cross-over probability $\epsilon = 0.001$.

4 points
Extra workspace
7. Draw a (recursive systematic) encoder for the rate $\frac{1}{2}$ convolutional code having generator matrix

$$G(D) = \begin{bmatrix} 1 & \frac{1+D}{1+D+D^2} \end{bmatrix}.$$ 

4 points
8. The identical figures on the next 3 pages show the trellis diagram of a convolutional code having

\[ G(D) = \begin{bmatrix} 1 + D + D^2 & 1 + D \end{bmatrix} \]

in which the trellis which begins at node level 0 and has been terminated at node level \( t = 4 \). The received signal over a binary symmetric channel having crossover probability \( \epsilon \ll 1 \) is as shown below:

\[ \{ (r_1(t), r_2(t)) \} = 01\ 11\ 11. \]

(a) Use the BCJR algorithm to determine the most likely third bit \( u_2 \). (The message sequence symbols are \( u_0, u_1, u_2, u_3 \)) As intermediate steps:

i. On the trellis marked (FORWARD) write in the appropriate locations the relevant \( \alpha_k(s_k) \) (or scaled versions of these) that you will need to implement the BCJR algorithm in this case 3 points

ii. On the trellis marked (BACKWARD) write in the appropriate locations the relevant \( \beta_{k+1}(s_{k+1}) \) (or scaled versions of these) that you will need to implement the BCJR algorithm in this case 3 points

iii. On the trellis marked backward, also write in the relevant \( \gamma_k(s_k, s_{k+1}) \) (or scaled versions of these) that you will need 3 points

iv. Compute the likelihood ratio

\[ Pr(u_k = 1 \mid \underline{r})/Pr(u_k = 0 \mid \underline{r}). \]

where \( \underline{r} \) denotes the received vector. 3 points

(b) Use the Viterbi algorithm and the trellis marked (Viterbi) to determine the most likely message sequence \( u_0, u_1, u_2, u_3 \). 4 points
Use this page to determine and show **FORWARD** recursion parameters
Use this page to determine and show **backward** recursion parameters.
Use this page to decode using the Viterbi algorithm.
Extra workspace

Enjoy your summer!