EE224A Fall 2009: Homework #8

Assigned: 11/19/2009 --- Due: 12/1/2009 in class *before* lecture

Problem 1 Problems in Praokis.

Problem 7.13

The generator matrix for a linear binary code is

$$\mathbf{G} = \begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix}$$

a Express **G** in systematic $[\mathbf{I} \mid \mathbf{P}]$ form.
b Determine the parity check matrix **H** for the code.
c Construct the table of syndromes for the code.
d Determine the minimum distance of the code.
e Demonstrate that the code word corresponding to the informatin sequence 101 is orthogonal to **H**.

Problem 7.25

Show that when a binary sequence $\boldsymbol{x}$ of length n is transmitted over a BSC with crossover probability p, the probability of receiving $\boldsymbol{y}$, which is at Hamming distance d from $\boldsymbol{x}$, is given by

$$P(\boldsymbol{y}|\boldsymbol{x}) = (1-p)^n (p/(1-p))^d$$

From this conclude that if $p < ½$, $P(\boldsymbol{y}|\boldsymbol{x})$ is a decreasing function of d and hence ML decoding is equivalent to minimum-Hamming-distance decoding. What happens if $p > ½$?

Problem 7.31

A $(k+1, k)$ block code is generated by adding 1 extra bit to each information sequence of length k such that the overall parity of the code (i.e., the number of 1s in each codeword) is an odd number. Two students, A and B, make the following arguments on *error detection* capability of this code.

1. Student A: Since the weight of each codeword is odd, any single error changes the weight to an even number. Hence, this code is capable of detecting any single error.

2. Student B: The *all-zero* information sequence 00…0 of length k will be encoded by adding one extra 1 to generate the codeword 00…01 of length $k + 1$. This means that there is at least one codeword of weight 1 in this code. Therefore, $d_{\min} = 1$, and since any code can detect at most $d_{\min} - 1$errors, and for this code $d_{\min} - 1 = 0$, this code cannot detect any errors.

   Which argument do you agree with and why? Give your explanation in one short paragraph.

Problem 2

In this homework, you will be asked to implement on Matlab the iterative decoding of Fountain codes. The main reference is the talk on Fountain codes by Amin Shokrollahi, which we have posted on the web. I suggest you go through those slides first before starting the problem. You should answer each part of the question, and hand in any Matlab code and plots.

(a) A binary linear code consists of K information bits and N coded symbols, each of which is a modulo 2 sum of a subset of the information bits. Design a data structure for representing the code. The data structure should be both space-efficient and easy to update by the iterative decoding algorithm. You can assume that the number of information bits that each coded symbol depends on is bounded by D, and D is much smaller than N and K (low-density).

(b) Implement the iterative decoding algorithm discussed in class and shown on the Fountain code slides. Test your algorithm on the simple example discussed in class: three information bits b_1, b_2, b_3. Six coded symbols: $x_1 = b_1, x_2 = b_2, x_3 = b_3, x_4 = b_1+b_2+b_3, x_5 = b_2+b_3, x_6 = b_1 + b_2$. The second and third coded symbols are erased.

(c) Now consider the fountain code, where each coded symbol is independently and randomly generated as described in the slides. You can use the following weight distribution (Ω_i is the probability that i information bits are combined to form the coded symbol):

$$\Omega_1 = 0.007969, \Omega_2 = 0.493570, \Omega_3 = 0.166220, \Omega_4 = 0.072646, \Omega_5 = 0.082558, \Omega_8 = 0.056058,$$

$$\Omega_9 = 0.037229, \Omega_{19} = 0.055590, \Omega_{65} = 0.025023, \Omega_{66} = 0.003135$$

Suppose $K = 5000$. And suppose the destination receives 100 such coded symbols. Run the iterative decoding algorithm. How many information bits can you decode before the algorithm stalls? Give an upper bound on the maximum possible number of bits that can be decoded from the 100 symbols.

(d) Now keep the 100 symbols already received and suppose you receive an additional 100 coded symbols. Run the iterative decoding algorithm on the 200 coded symbols. How many information bits can be decoded, and compare to the upper bound.

(e) Repeat part (d), adding 100 more symbols each time, until all the 5000 information are decoded. Plot the number of information bits decoded as a function of the number of received coded symbols, always comparing it to the upper bound. Comment on the performance of the algorithm, both in terms of the progress it makes as it receives more coded symbols, and in terms of the number of coded symbols needed to decode all the bits.