Problem Set 4  

Date issued: April 20, 2004  
Date Due: April 27, 2004  

Reading Assignment: Finish Reading Chap. 5 (Sections 5.1-5.8)  

Reminder: The midterm exam will be on Tuesday, May 4.

Do the following problems:

1. Consider a source alphabet with 4 letters \{a_1, a_2, a_3, a_4\} and corresponding probabilities \(p_1 \geq p_2 \geq p_3 \geq p_4\). For this source, consider a Huffman code that encodes a single letter at a time.
   a. Find all possible values of \(p_1 \geq p_2 \geq p_3 \geq p_4\) such that both (2,2,2,2) and (1,2,3,3) are optimal sets of codeword lengths (i.e. have the minimum average length).
   b. Can any other set of codeword lengths be optimal?
   c. Is there a set of probabilities \(p_1 \geq p_2 \geq p_3 \geq p_4\) such that the minimum expected codeword length for the source is greater than 2?

2. Problem 5.8 in C&T.

3. Run-Length Coding: A \(k\)-bit run-length code is a variable-to-fixed source code that parses a sequence from a binary source into runs of identical symbols of length at most \(2^k - 1\).
   - If a run of length less than \(2^k\) is found, its length is encoded using \(k\) bits and the run is deleted from the source.
   - If a run of length greater than or equal to \(2^k\) is found, \(k\) zero bits are emitted and \(2^k - 1\) bits are deleted from the source.

For example if \(k=2\) the sequence 0011111100 is encoded as 10001110.

Assume the binary source \(X_1, X_2, \ldots\) is a simple Markov chain with \(\Pr(X_n \neq X_{n-1}) = \varepsilon\), and that the first source symbol is always 0.
   a. Find the entropy rate of the source as a function of \(\varepsilon\).
   b. Find the compression ratio (number of output bits per input bit) as a function of \(\varepsilon\) and \(k\).
   c. For \(\varepsilon = .1\) numerically find the best compression ratio achieved and compare this to the entropy rate. Repeat for \(\varepsilon = .01\).
   d. Why might this type of code be used instead of a Huffman code?

4. Problem 5.12 in C&T

5. Problem 5.24 in C&T (Hint: in part a., give an iterative rule for constructing the set \(S_i\)).