Boğaziçi University
Department of Physics

Phys 497 Spring 2008

Problem Set #2

Due in class Tuesday, 6 May 2008

Problem 1:
Invent a lossless compression method, and answer the following questions:

a) Describe the compression algorithm. *(10 points)*

b) Describe the decompression algorithm. *(10 points)*

c) Give an example where the input is *actually* compressed. *(10 points)*

d) Given an example where the input is *expanded.(10 points)*

Problem 2: *(30 points)*
The intent of this question is to solidify your understanding of the process of error detection and correction.

The essential operation in error handling is that we take k data bits and add a number of redundancy bits (which may or may not be parity bits) to produce codewords of length n. (Thus we add $n - k$ extra bits.)

One of the key points is the hamming distance of two valid codewords to each other. If there are two valid codewords at hamming distance one to each other, we can not detect single errors. A hamming distance of one means that a single error will convert one valid codeword into another valid codeword, making it impossible to detect errors.

If the hamming distance between any two valid codewords is two, then we can detect single errors. This is the case of using a single parity bit per codeword. In this case, correction is not possible, since a codeword with a single error is at hamming distance one from multiple valid codewords.

If the hamming distance between any two valid codewords is three, then we can correct single errors. However, double errors will mimic single errors. This is the case with Hamming Codes.

Consider the case of using three bits plus one parity bit using codewords of four bits in total for parts **a** and **b**.

a. Write down all 16 possible codewords. Mark the eight valid codewords.

b. Now, consider each of the eight invalid codewords. How many valid codewords are there at hamming distance one for each invalid codeword?

Now, consider the 7-bit hamming code discussed in class for parts **c** and **d**.

c. Since we have four data bits, there are a total of 16 valid codewords. List them.

d. For each valid codeword, list all the invalid codewords at a hamming distance one to it. Verify that, once you have done this, all codewords are listed, and no codeword is listed more than once.

e.Consider a code where the hamming distance between two valid codewords is three. What sorts of errors can you detect? What sorts of errors can you correct?

f. If you want to be able to correct double errors, what is the minimum hamming distance between any two valid codewords in your code?

Problem 3: *(20 points)*
Assume you are to transmit 255-bit blocks.

a. If you use a Hamming code, what is your code rate? (You can just look this up in the notes.)

b. What is the best code rate you can get with a rectangular code?

Problem 4: *(20 points)*
If we only wanted to *detect* single errors, we could use block codes with a single parity bit. If your block size was n, only one bit would be used for parity, so your efficiency (code rate) would be:

$$\frac{n-1}{n} = 1 - \frac{1}{n}$$

Note that here, any block size is possible. When you want to *correct* single errors, and detect double errors, the best you get is Hamming codes. Here, of course, not every block size is possible. But, it is possible to derive a formula for the efficiency of Hamming codes as a function of **block size** n. Find out what it is.