Tony Majestro

Problem Set 8

1. You have received the following seven-bit strings, in which each string represents a four-bit message encoded using the Hamming (7,4) single-error correcting code:

0101011 0001011 0110110 1101111 0001001 0010110

1001111 0011110 0111010 1111010 1110100

1. Assuming that at most one error was made in transmitting each string, decode these strings and exhibit the sequence of four-bit messages corresponding to the strings. For each string, tell which bit, if any, was incorrectly received.

To decode each message, we can find values for alpha, beta, and gamma, where the concatenation of alpha, beta, and gamma represents the binary representation of the error bit position. Alpha can be found by finding the XOR sum of bits 4, 5, 6, and 7. Beta can be found by finding the XOR sum of bits 2, 3, 6, and 7. Gamma can be found by finding the XOR sum of bits 1, 3, 5, and 7.

This can be decoded using the following python program:



This gives the following error positions and corrected codes:

|  |  |  |
| --- | --- | --- |
| Original Code | Corrected Code | Error Position |
| 0101011 | 0101010 | 7 |
| 0001011 | 0001111 | 5 |
| 0110110 | 0010110 | 2 |
| 1101111 | 1111111 | 3 |
| 0001001 | 0011001 | 3 |
| 0010110 | 0010110 | 0 |
| 1001111 | 0001111 | 1 |
| 0011110 | 0010110 | 4 |
| 0111010 | 0101010 | 3 |
| 1111010 | 1011010 | 2 |
| 1110100 | 1110000 | 5 |

Since the Hamming (7, 4) code has the message bits in the 3rd, 5th, 6th, and 7th positions of the original message, we can extract these bits from the corrected codes to get the corrected messages:

0010 0111 1110 1111 1001 1110

0111 1110 0010 1010 1000

1. Each of the four-bit message strings corresponds to an integer between 0 and 15, with leading zeros prepended to make the string four bits long. (Thus, 5 is 101 in binary, and so corresponds to the string 0101.) Exhibit the sequence of integers corresponding to the sequence of four-bit messages from part (a).

The previous corrected 4-bit binary messages can be expressed as the following base-10 numbers:

2 7 14 15 9 14 7 14 2 10 8

1. The correspondences 0=A, 1=B, 2=C, 3=D, 4=E, 5=F, 6=G or Q, 7=H or R, 8=I or S, 9=J or T, 10=K or U, 11=L or V, 12=M or W, 13=N or X, 14=O or Y, and 15=P or Z, properly applied to the sequence of integers in part (b), will spell out a meaningful phrase. What is that that phrase?

The previous base 10 numbers can be converted to the follow letter(s):

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 2 | 7 | 14 | 15 | 9 | 14 | 7 | 14 | 2 | 10 | 8 |
| c | h | O | p | j | o | h | o | c | k | I |
|  | r | Y | z | t | y | r | y |  | u | s |

The first word could be chop, crop, or crypto. If we guess the first word to be crypto, then the second word must be rocks. This is more likely than the phrase ‘chop to rocks’ or ‘crop to rocks’. Therefore, the phrase is probably ‘crypto rocks’, since this is a cryptography class.

1. Use the four-bit correspondences described in Problem 1, parts (b) and (c), to encoded the plaintext message “TECH” using the Hamming (7,4) single-error correcting code.

Using the bit-correspondences from part 1, we can write the message “TECH” as four 4-bit numbers:

T E C H

9 4 2 7

1001 0100 0010 0111

We can then rewrite each message as a code without the error correcting bits filled in:

9: \_ \_ 1 \_0 0 1

4: \_ \_ 0 \_ 1 0 0

2: \_ \_ 0 \_ 0 1 0

7: \_ \_ 0 \_ 1 1 1

In the Hamming (7, 4) code, the 4th bit is chosen so that x4 + x5 + x6 + x7 mod 2 = 0. This means that x4 is equal to x5 + x6 + x7. For the message that encodes 9, this gives 0 + 0 + 1 mod 2 = 1. Therefore, the 4th bit is 1:

9: \_ \_ 1 **1** 0 0 1

To encode the 2nd bit, we choose x2 so that x2 + x3 + x6 + x7 mod 2 = 0. This means x2 = x3 + x6 + x7. For the message that encodes 9, this gives 1 + 0 + 1 mod 2 = 0. Therefore, the 2nd bit is 0:

9: \_ **0** 1 1 0 0 1

To encode the 1st bit, we choose x1 so that x1 + x3 + x5 + x7 mod 2 = 0. This means x1 = x3 + x5 + x7. For the message that encodes 9, this gives 1 + 0 + 1 mod 2 = 0. Therefore, the 1st bit is 0:

9: **0** 0 1 1 0 0 1

This process can be repeated for the messages 4, 2, and 7:

4: \_ \_ 0 \_ 1 0 0

4: \_ \_ 0 **1** 1 0 0

4: \_ **0** 0 1 1 0 0

4: **1** 0 0 1 1 0 0

2: \_ \_ 0 \_ 0 1 0

2: \_ \_ 0 **1** 0 1 0

2: \_ **1** 0 1 0 1 0

2: 0 1 0 1 0 1 0

7: \_ \_ 0 \_ 1 1 1

7: \_ \_ 0 **1** 1 1 1

7: \_ **0** 0 1 1 1 1

7: **0** 0 0 1 1 1 1

This gives the following Hamming (7, 4) codes for the message “TECH”:

0011001 1001100 0101010 000111

1. If s and t are finite strings (e.g. character strings, number strings, bit strings), define the Hamming distance H(s, t) to be the number of positions at which s and t differ. Prove that (a) H(s, t) >= 0, with equality if and only if s = t; (b) H(s, t) = H(t, s), and (c) for all strings s, t, and u, H(s, u) <= H(s, t) + H(t, u).
2. If s is equal to t, then every value in every position of their respective strings are the same. Therefore, since they have no positions that differ, their Hamming distance is 0. Also, if two strings have a Hamming distance of 0, then they have no positions in their strings that are different. Therefore, their strings must be the same.

Thus, the Hamming distance H(s, t) = 0 if and only if s == t.

1. The Hamming distance H(s, t) is equal to the number of positions at which s and t differ. Also by definition, H(t, s) is the number of positions at which s and t differ. Therefore, the number of positions at which t differs from s is equal to the number of positions at which s and t differ. Therefore, H(s, t) = H(t, s).
2. Proof by contradiction:

Assume that for all strings s, t, and u, that H(s, t) + H(t, u) < H(s, u). By definition of Hamming distance, we can change the string s into the string t into H(s, t) moves by modifying each position of s that differs from t.

Then we can get from s to t in H(s, t) moves. We can also get from t to u in H(t, u) moves. This means that the shortest path from string s to the string u takes H(s, t) + H(t, u) moves, and therefore H(s, u) = H(s, t) + H(t, u).

However, this is a contradiction, because we assumed that H(s, t) + H(t, u) < H(s, u). Therefore, the assumption is false, and H(s, u) <= H(s, t) + H(t, u).