Massachusetts Institute of Technology
Department of Electrical Engineering and Computer Science

6.02 **Solutions to Chapter 5** Updated: February 16, 2012

Please send information about errors or omissions to hari; questions are best asked on piazza.

1. The Hamming distance of a n-bit word is the sum of the Hamming distances for every single bit of the n-bit word. Listing all of the possible combinations of x, y, and z for a single bit (note: due to symmetry, only four of the possible combinations are relevant):

x	y	z	$\text{HD}(x,y) + \text{HD}(y,z) \geq \text{HD}(x,z)$
0	0	0	$0 + 0 \geq 0$
1	0	0	$1 + 0 \geq 1$
1	1	0	$0 + 1 \geq 1$
1	1	1	$0 + 0 \geq 0$

   We find that the triangle inequality holds for each combination. Since the Hamming distance satisfies the triangle inequality for every bit, it satisfies the triangle inequality for the entire n-bit word.

2. $(23, 15, 3)$
   n = number of data bits + number of parity bits = $15 + 8 = 23$
   k = number of data bits = 15
   $d = 3$. See section 5.5 for a discussion of why $d = 3$.

3. (a) It is not possible to obtain a rate lower than 1/3. In order to achieve this rate, each codeword will consist of a single data bit and 2 parity bits. If we increase the number of data bits in each codeword, we obtain a rate that is larger than 1/3; and in order to decrease the rate, we would need to have a codeword that consisted of less than 1 data bit, which is not possible.

   (b) We would like to see whether rectangular codes exist whose rates are $1/2, 2/3, \ldots, {}^{\ell-1}/_{\ell}$. We know that the rate of a rectangular code with r rows and c columns is $\frac{rc}{rc+r+c}$.

$$\begin{aligned} \frac{rc}{rc+r+c} &= \frac{\ell-1}{\ell} \\ \Rightarrow rc - \ell r - \ell c &= 0 \\ \Rightarrow rc - \ell r - \ell c + \ell^2 &= \ell^2 \\ \Rightarrow (r-\ell)(c-\ell) &= \ell^2 \end{aligned}$$

   If we set $r - \ell = \ell$ and $c - \ell = \ell$, the equation is satisfied, which means that a rectangular code with parameters $(4\ell^2 + 4\ell, 4\ell, 3)$ has rate ${}^{\ell}/_{\ell+1}$ (there may be other solutions for particular values of ℓ, e.g., when ℓ is not a prime number). This is an interesting observation because although the number of parity bits in a rectangular code grows at least as fast as the square-root of the number of message bits (ℓ), it is still possible to achieve "high" code rates of the form ${}^{\ell}/_{\ell+1}$.

Hari Balakrishnan

4. (a) $(5, 2)$
   n = number of data bits + number of parity bits = $2 + 3 = 5$
   k = number of data bits = 2

   (b) $E_0 = (D_0 + P_0) \bmod 2$
   $E_1 = (D_0 + D_1 + P_1) \bmod 2$
   $E_2 = (D_1 + P_2) \bmod 2$

   (c) The syndrome table is:

$E_2E_1E_0$	Corrective Action
000	no errors
001	single error (P_0 has an error, flip to correct)
010	single error (P_1 has an error, flip to correct)
011	single error (D_0 has an error, flip to correct)
100	single error (P_2 has an error, flip to correct)
101	multiple errors (unable to correct)
110	single error (D_1 has an error, flip to correct)
111	multiple errors (unable to correct)

   Observe that the number of syndrome table entries corresponding to the "no error" or "single correctable error" case is $n + 1 = 6$.

   (d) 01011. $E_0 = 1$, $E_1 = 1$, $E_2 = 0$ which corresponds to a single error in D_0 from table in the solution of part (c). The receiver can then flip D_0 to obtain the corrected codeword 01011.

   (e) No such code is possible. In order to have single-bit error correction, the bound $2^{n-k} \geq n + 1$ must be satisfied ($2^{n-k} = 4 \ngeq n + 1 = 5$).

5. (a) (i) Code rate = $k/n = 1/2$
   (ii) 3. All minimum weight, non-zero codewords are $D_1D_2D_3P_1P_2P_3$ = 100101,010110,001011,111000.
   (iii) Hamming distance = 3. Since the code is a linear block code, Theorem 6.5 applies and states that the minimum Hamming distance is equal to the weight of the non-zero codeword with smallest weight.

6. The table is:

$E_3E_2E_1$	Error Pattern
000	no errors
001	single error (P_1 has an error, flip to correct)
010	single error (P_2 has an error, flip to correct)
011	single error (D_2 has an error, flip to correct)
100	single error (P_3 has an error, flip to correct)
101	single error (D_1 has an error, flip to correct)
110	single error (D_3 has an error, flip to correct)
111	multiple errors (unable to correct)

7. The addition of the extra parity bit increases the minimum Hamming distance from 3 to 4, but the extra parity bit has no effect on the error correction capability. As a result, the code

can detect up to 3 bit errors while it can only correct 1 bit error. A simple example why the error correction capability is not increased can be found when considering the new syndrome $E_1E_2E_3E_4$, where E_1, E_2, and E_3 are defined in the problem 5 and $E_4 = \sum_{i=1}^{3} D_iP_i + P_4$. Assume that the calculated syndrome is 0110. This syndrome is generated for the following bit errors: D_1D_2, D_3P_4, and P_2P_3. As a result, the code can detect any three bit errors, but it cannot correct them because of the multiple correction possibilities. To show that the minimum Hamming distance is 4, consider errors in the bits D_1, D_2, D_3, and P_4. The syndrome for this case will be 0000 indicating that no error occurred. Any other combination of 4 bit errors will be detected by the code since the calculated syndrome will be non-zero. In addition, any combination of 1, 2, or 3 bit errors will also result in a non-zero syndrome.

8. (a) The code is a linear block code because the sum of any two codewords is another codeword. The rate is $k/n = 2/3$.

   (b) The code is a linear block code because the sum of any two codewords is another codeword. The rate is $k/n = 2/3$.

   (c) The code is *not* a linear block code because the sum of 111 and 100 is 011, which is not a codeword.

   (d) The code is a linear block code because the sum of any two codewords is another codeword. The rate is $k/n = 2/5$.

   (e) The code is a linear block code since the sum of 00000 with 00000 is equal to 00000. In this case, $n = 5$ but $k = 0$ so the code rate is 0 (i.e., the receiver already knows what is sent so no information is transferred).

9. An (n,k) block code can represent in its parity bits at most 2^{n-k} patterns that must cover all of the error cases we wish to correct, as well as the one case with no errors. When the minimum Hamming distance is $2t+1$, the code can correct up to t errors. The number of ways in which the transmission can experience $0, 1, 2, \ldots, t$ errors is equal to $1 + \binom{n}{1} + \binom{n}{2} + \cdots + \binom{n}{t}$. This number must not exceed 2^{n-k} because the maximum number of expressible syndromes is 2^{n-k}, which proves the assertion.

10. We can verify if a specific code exists if the bound $n + 1 \leq 2^{n-k}$ (for single error correction or derivations of this bound for a larger number of error corrects) is satisfied.

    (a) YES. $\left\lfloor \frac{D-1}{2} \right\rfloor = 1$, $n + 1 = 32 \leq 2^{n-k} = 2^{31-26} = 2^5 = 32$. There are enough parity bits to ensure that a minimum Hamming distance of 3 is possible.

    (b) NO. $\left\lfloor \frac{D-1}{2} \right\rfloor = 1$, $n + 1 = 33 \leq 2^{n-k} = 2^{32-27} = 2^5 = 32$. There are no linear block codes that can correct a single error, so $(32, 27, 3)$ is not a possible code.

    (c) YES. The simple parity code (adding all the bits in the message together so that each codeword has an even number of ones) is a $(43, 42, 2)$ code; in general, simple parity is a $(n + 1, n, 2)$ code for any $n \geq 1$.

    (d) YES. $(27, 18, 3)$ are the parameters. First, note that $\left\lfloor \frac{D-1}{2} \right\rfloor = 1$, $n + 1 = 28 \leq 2^{n-k} = 2^{27-18} = 2^9$, so there seem to be enough parity bits to construct such a code. The rectangular code with parameters $r = 6, c = 3$ gives us $n = rc + r + c = 27$ and $k = rc = 18$, and we know that any rectangular code with $r, c > 1$ has Hamming distance 3.

    (e) Very similar to a PSet problem; see PSet solutions after the due date!

11. The (15,11) code can be constructed as follows:

index	1	2	3	4	5	6	7	8
binary index	0001	0010	0011	0100	0101	0110	0111	1000
(15,11) code	p_1	p_2	d_1	p_3	d_2	d_3	d_4	p_4

9	10	11	12	13	14	15
1001	1010	1011	1100	1101	1110	1111
d_5	d_6	d_7	d_8	d_9	d_{10}	d_{11}

The above construction shows that there are four parity bits (or equations) where 7 message bits contribute to each parity bit. For example, the binary index indicates that the parity check equation for p_1 is $p_1 = d_1 + d_2 + d_4 + d_5 + d_7 + d_9 + d_{11}$, which contains 7 message bits. The rest of the parity check equations yield similar results.

12. See the solution for problem 1 for a proof of Theorem 6.2.

Theorem 6.3 may be established as follows. A maximum likelihood decoder maximizes the quantity $\mathbb{P}(r|c)$; i.e., it finds c so that the probability that r was received given that c was sent is maximized. Consider any codeword $\tilde{c}$. For a BSC with error probability p_e, if r and $\tilde{c}$ have a Hamming distance of d, then $\mathbb{P}(r|c) = p_e^d(1-p_e)^{N-d}$, where N is the length of the received codeword (and also the length of each valid codeword). It's more convenient to take the logarithm of this conditional probaility, also termed the *log-likelihood*:[1]

$$\log \mathbb{P}(r|\tilde{c}) = d \log p_e + (N-d)\log(1-p_e) = d \log \frac{p_e}{1-p_e} + N \log(1-p_e). \quad (1)$$

If $p_e < 1/2$, then $\frac{p_e}{1-p_e} < 1$ and the log term is negative (otherwise, it's non-negative). As a result, minimizing the log likelihood boils down to minimizing d, because the second term on the RHS of Eq. (1) is a constant. This completes the proof of Theorem 6.3.

13. By definition, the sum of any two codewords in a linear block code is also a codeword. We will first consider the case when we have all even weight codewords. Since the sum of any two even weight codewords is also an even weight codeword, it is possible to have a linear block code that consists of only even weight codewords. Now consider the case where we have all odd weight codewords. Since the sum of any of these codewords must also be a codeword and the sum of two odd weight codewords is an even codeword, all possible linear combinations of these codewords will result in an equal number of even and odd weight codewords. This shows that any linear block code must either have only even weight codewords, or have an equal number of even and odd weight codewords.

14. First assign "0" = "red" and "1" = "blue". Once everyone has formed a line, start from the back of the line (i.e., the tallest person) and have that person say the sum (modulo 2) of all hat colors in front of him or her (i.e., the overall parity). The next-tallest person in line will then take the number (or color in this case) that the person behind them said and add that to the sum of all hat colors in from of him or her, all additions being done modulo 2. The result of this sum is the color of their hat, which they yell out to the next person in line. This continues until everyone has yelled out the colors of their hats. The only person that may yell out an

[1] The base of the logarithm doesn't matter to us at this stage, but traditionally the log likelihood is defined as the natural logarithm (base e).

incorrect hat color is the tallest person in line who made the first announcement (because that person announced the overall parity), which would give the team a score $\geq N - 1$.