

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

School of Computer and Communication Sciences

Handout 31

Solutions to Homework 12

Information Theory and Coding

Dec. 20, 2021

PROBLEM 1.

- (a) Since C is non-empty, it contains some codeword x . By linearity C must contain $x + x$. But, for any x , $x + x$ is the all-zero sequence since we are doing modulo-2 sums. So, C contains the all-zero sequence.
- (b) The elements of D' are those sequences of the form $x + y$ where y is in D . Since x is in C and D is a subset of C , any x and y are both in C , and so is their sum.
- (c) Suppose there was an element z common to D and D' . Then $z = x + y$ where y is in D . Since we assumed that D is a linear subset, then $z + y$ is also in D . But $z + y$ equals x , and we arrive at the contradiction that x is in D .
- (d) Since the mapping $y \mapsto x + y$ is a bijection, D and D' are in one-to-one correspondence, and hence have the same number of elements.
- (e) Suppose z_1 and z_2 are in $D \cup D'$. There are four possibilities: (1) both z_1 and z_2 are in D , (2) both z_1 and z_2 are in D' , (3) z_1 is in D , z_2 is in D' , (4) z_1 is in D' , z_2 is in D . In case (1), the linearity of D implies that $z_1 + z_2$ is in D . In case (2), $z_1 = x + y_1$ and $z_2 = x + y_2$ for some y_1 and y_2 both in D , then $z_1 + z_2 = x + x + y_1 + y_2 = y_1 + y_2$ is in D . In case (3) $z_2 = x + y_2$ and $z_1 + z_2 = x + (z_1 + y_2)$, which is in D' , and similarly in case (4). Thus in all cases $z_1 + z_2$ is in $D \cup D'$ and we see that $D \cup D'$ is a linear subset of C .
- (f) We thus see that if at the beginning of step (ii) D is a linear subset of C , at the end of step (iii) $D \cup D'$ is linear, is a subset of C because both D and D' are, and has twice as many elements of D since D' has the same number of elements of D and is disjoint from it. Thus, when the algorithm terminates, D contains all elements of C and since it is a subset of C it must equal C . Furthermore, its size, being equal to successive doublings of 1, is a power of 2.

PROBLEM 2.

Let x and x' be two different codewords in the extended Hamming code. Let z and z' be the parts of x and x' that come from the Hamming code (i.e., z is all but the last bit of x , and z' that of x'), and p and p' be the bits appended to z and z' to get x and x' . Since x and x' are different then so are z and z' : if $z = z'$ then $p = p'$ and x and x' would have been the same. Thus, $d_H(z, z') \geq 3$ since z and z' are different Hamming codewords. On the other hand, if $d_H(z, z') = 3$, then z and z' must have different parity: if they both had an even number of 1's or both had an odd number of 1's they would have differed in an even number of places and $d_H(z, z')$ would have been an even number. Thus, if $d_H(z, z') = 3$ then $p \neq p'$ and we have $d_H(x, x') = 4$. If $d_H(z, z') \geq 4$ then clearly $d_H(x, x') \geq 4$. We thus see that the minimum distance of the new code is 4.

Consider the following procedure to decode

Given a sequence y , compare it to all the codewords and find the number of positions in which y differs from them. If there is a unique codeword for which this number is smallest, declare that codeword. If not, declare ‘errors were detected’.

If the minimum distance d of a code is an even number, $d = 2j$, then if a sequence y differs from the transmitted codeword x by up to $j - 1$ places, then y will be closer to the transmitted codeword than to any other and the decoder will correctly decode x . If however, y differs from x in j places, then no other codeword will be closer to y , but there might be a codeword x' which also differs from y in j places. In such a case the decoder will not be able to correct but detect the errors. In particular, if $d = 4$, then all single errors are corrected and all double errors are detected (may even be corrected).

PROBLEM 3.

- (a) Suppose \mathbf{x} and \mathbf{x}' are two codewords in \mathcal{C} . Then for $\forall i = 0, 1, \dots, m - 1$,

$$\begin{aligned}x_0 + x_1\alpha_i + \dots + x_{n-1}\alpha_i^{n-1} &= 0 \\x'_0 + x'_1\alpha_i + \dots + x'_{n-1}\alpha_i^{n-1} &= 0\end{aligned}$$

Therefore,

$$(x_0 + x'_0) + (x_1 + x'_1)\alpha_i + \dots + (x_{n-1} + x'_{n-1})\alpha_i^{n-1} = 0 \quad \text{for } \forall i = 0, 1, \dots, m - 1.$$

which shows $\mathbf{x} + \mathbf{x}'$ is also a codeword.

- (b) $x(D) = x_0 + x_1D + \dots + x_{n-1}D^{n-1}$ is a polynomial of degree (at most) $n - 1$ and (x_0, \dots, x_{n-1}) is a codeword if $\alpha_0, \alpha_1, \dots, \alpha_{m-1}$ are m of its roots. This means

$$x(D) = (D - \alpha_0)(D - \alpha_1) \dots (D - \alpha_{m-1})h(D) = g(D)h(D)$$

for some $h(D)$. Note that $h(D)$ can have degree (at most) $n - m - 1$. On the other side, there is a one-to-one correspondence between the codewords of \mathcal{C} and degree $n - 1$ polynomials. Since $g(D)$ is fixed for all codewords, a polynomial $x(D)$ corresponding to a codeword \mathbf{x} is determined by choosing the coefficients of $h(D) = h_0 + h_1D + \dots + h_{n-m-1}D^{n-m-1}$. Since $h_j \in \mathcal{X}$ for $j = 0, 1, \dots, n - m - 1$ we have q^{n-m} different $h(D)$ s and, thus, q^{n-m} codewords.

- (c) For every column vector $\mathbf{u} = [u_0, u_1, \dots, u_{m-1}]^T$, $A\mathbf{u} = [u(1), u(\beta), \dots, u(\beta^{n-1})]^T$. Consequently, $A\mathbf{u} = \mathbf{0}$ means $u(D)$ has n roots which is impossible (since it is a polynomial of degree $m - 1 < n$).
- (d) Using the same reasoning as in (c) one can verify that $\mathbf{x} = (x_1, \dots, x_n)$ is a codeword iff $\mathbf{x}A = \mathbf{0}$. This means A is the parity-check matrix of the code \mathcal{C} . Since the code is linear, using Problem 4 of Homework 11 we know that has minimum distance d iff every $d - 1$ rows of H are linearly independent and some d rows are linearly dependent. That A has rank m implies there are no m linearly dependent rows thus $d \geq m + 1$. On the other side, we know from the Singleton bound that a code with q^{n-m} codewords and block-length n has minimum distance $d \leq m + 1$. Thus we conclude that $d = m + 1$.

PROBLEM 4.

- (a) As H had four columns the blocklength $n = 4$. Observe that we can rearrange $H\mathbf{x} = \mathbf{0}$ to solve for x_1, x_2 in terms of x_3, x_4 . As there are 3^2 possibilities for (x_3, x_4) the code has $M = 9$ codewords. The code rate is thus $\frac{1}{2} \log 3$.
- (b) The receiver receives $\mathbf{y} = \mathbf{x} + \mathbf{z}$ where \mathbf{z} is either the zero vector, or it has only a single nonzero component z_i which can take the value 1 or 2. With h_i denoting the i th column of H , $H\mathbf{y} = H\mathbf{z}$ is either zero, or takes on the value h_i (if $z_i = 1$) or $2h_i$ ($z_i = 2$). Since the collection of eight vectors $h_1, 2h_1, h_2, 2h_2, h_3, 2h_3, h_4, 2h_4$ are all distinct and different from zero, the receiver can identify if z is the zero vector or the i and the value of z_i from $H\mathbf{y}$
- (c) This will increase the block length to 5 and the number of codewords to 3^3 yielding a new rate of $\frac{3}{5} \log 3$ which is larger than the rate found in (a).
- (d) We need to ensure that the new column and its multiple by 2 is different from the zero and the collection of 8 vectors above. We see that this is not the case for any of the vectors listed.
- (e) Now z_i can take on only the value 1 (but not 2). Thus to ensure detection and correction we only need h_i 's to be distinct and different from zero. Now, all columns except the zero column in (d) can be added.