

Queen Mary
UNIVERSITY OF LONDON
MAS309 Coding Theory

Tuesday 17th May 2005
2:30 p.m.

Duration: 2 hours.

You may attempt as many questions as you wish and all questions carry equal marks. Except for the award of a bare pass, only the best four questions will be counted.

Electronic calculators are not permitted.

1. Let n and d be integers greater than 1.
 - (a) Suppose \mathcal{C} is a binary (n, M, d) -code. Show how to construct a binary $(n-1, M, d-1)$ -code from \mathcal{C} , and prove that it really is an $(n-1, M, d-1)$ -code. Deduce that $A_2(n-1, d-1) \geq A_2(n, d)$.
 - (b) Now suppose that d is even. If \mathcal{D} is a binary $(n-1, M, d-1)$ -code, explain how to construct a binary (n, M, d) -code from \mathcal{D} , and prove that it really is an (n, M, d) -code.
 - (c) Deduce that if d is even, then $A_2(n, d) = A_2(n-1, d-1)$.
 - (d) Using the Plotkin bound or otherwise, show that $A_2(n, d)$ is not necessarily equal to $A_2(n-1, d-1)$ if d is odd.
2. For this question, you may assume any basic linear algebra, including the Rank-Nullity Theorem.
 - (a) Explain what is meant by a *linear* $[n, k]$ -code over \mathbb{F}_q .
 - (b) Explain what is meant by a *generator matrix* for a linear $[n, k]$ -code.
 - (c) If \mathcal{C} is a linear $[n, k]$ -code over \mathbb{F}_q , explain what is meant by the *dual code* \mathcal{C}^\perp .
 - (d) If G is a generator matrix for \mathcal{C} and $v \in \mathbb{F}_q^n$, write down a criterion involving G for v to lie in \mathcal{C}^\perp .
 - (e) Deduce that \mathcal{C}^\perp is a linear $[n, n-k]$ -code.
 - (f) Prove that $(\mathcal{C}^\perp)^\perp = \mathcal{C}$.
 - (g) Explain what is meant by saying that a generator matrix is in *standard form*. Define what is meant by a *parity-check matrix*. If G is a standard form generator matrix for \mathcal{C} , describe how to find a parity-check matrix for \mathcal{C} .
 - (h) Write down a binary linear $[5, 2, 3]$ -code \mathcal{C} , and find a generator matrix and a parity-check matrix for \mathcal{C} . Find a generator matrix and a parity-check matrix for the Hamming code $\text{Ham}(2, 5)$.

© Queen Mary, University of London 2005

3. (a) Say what it means for a code to be *t-error-correcting*, and prove that a code \mathcal{C} is *t-error-correcting* if and only if the minimum distance of \mathcal{C} is at least $2t + 1$.
- (b) State the Hamming bound for *t-error-correcting* codes, and say what it means for a code to be *perfect*.
- (c) Show how to construct the binary Hamming code $\text{Ham}(r, 2)$, and prove that it is a perfect 1-error-correcting code. (You may assume any basic facts about linear codes which you state clearly.) Write down a parity-check matrix for $\text{Ham}(3, 2)$.

4. Suppose \mathcal{C} is a linear code over \mathbb{F}_q , with generator matrix G .

- (a) Say what it means for two linear codes to be *equivalent*.
- (b) Prove that if \mathcal{C} and \mathcal{D} are equivalent linear codes and w is an integer, then the number of words of weight w in \mathcal{C} equals the number of words of weight w in \mathcal{D} .
- (c) Describe five elementary matrix operations we may perform on a generator matrix to get a generator matrix for a code equivalent to \mathcal{C} .
- (d) If \mathcal{C} is a $[3, 2]$ -code, show that by using these five matrix operations, we may transform G into one of the matrices

$$\begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix}, \quad \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \end{pmatrix}, \quad \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}.$$

- (e) Deduce that if \mathcal{C} is a $[3, 2]$ -code with minimum distance 2, then \mathcal{C} is equivalent to the code with generator matrix

$$\begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix}.$$

(You may assume any results from the course about matrices, standard form and minimum distance, as long as you state them clearly.)

5. Suppose \mathcal{C} is a linear code of length n over \mathbb{F}_q .

- (a) Define the terms *coset* and *coset leader*, and describe how to construct a *Slepian array* for \mathcal{C} .
- (b) Prove that every word in \mathbb{F}_q^n appears in a Slepian array, and explain how to construct a nearest-neighbour decoding process for \mathcal{C} from a Slepian array for \mathcal{C} .
- (c) Write down a Slepian array for the binary code

$$\{00000, 01101, 10110, 11011\}.$$

- (d) Explain what a *parity-check matrix* for a linear code is, and what the *syndrome* of a word is. Define a *syndrome look-up table*, and explain how it is used to construct a nearest-neighbour decoding process for \mathcal{C} .
- (e) Write down a generator matrix and a parity-check matrix for the binary code

$$\mathcal{D} = \{00000, 10101, 01010, 11111\},$$

and construct a syndrome look-up table for \mathcal{D} .

(You do not have to explain your method, but doing so may help you to gain marks if you make arithmetic errors.)

6. In this question, we work with binary codes over the alphabet $\{0, 1\}$.

- (a) Using the matrix

$$\begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 \end{pmatrix}$$

or otherwise, find a binary $(7, 8, 4)$ -code, briefly explaining your construction.

- (b) Suppose \mathcal{C} is a binary (n, M, d) -code and m is a positive integer. Show how to construct a binary (nm, M, dm) -code, and prove that it really is a binary (nm, M, dm) -code.
- (c) State the Plotkin bound. (You should state both cases: d even and d odd.)
- (d) Prove that $A_2(7m, 4m) = 8$ for all positive integers m .
- (e) Prove that $4 \leq A_2(7m - 1, 4m) \leq 6$ for all positive integers m .

7. Suppose \mathcal{C} is a code over an alphabet A .

- (a) Explain what is meant by a *nearest-neighbour decoding process* for \mathcal{C} .
- (b) Suppose A is the binary alphabet $\{0, 1\}$, and \mathcal{C} is the code

$$\{0000, 0111, 1011, 1101, 1110\}.$$

Construct a nearest-neighbour decoding process for \mathcal{C} .

- (c) Suppose we send the word 0111 along a binary symmetric channel with symbol error probability $\frac{1}{4}$. For your chosen decoding process, calculate the word error probability.
 - (d) Define the *rate* of a binary (n, M, d) -code, and the *capacity* of a binary symmetric channel. State Shannon's Theorem.
 - (e) Define the *binary repetition code* of length n . Show that it has a unique nearest-neighbour decoding process if and only if n is odd.
8. For the following values of q, n, M, d , state (with reasons) whether a q -ary (n, M, d) -code exists. You may use any bounds from lectures that you state clearly.
- (a) $q = 2, n = 12, M = 52, d = 5$.
 - (b) $q = 3, n = 4, M = 9, d = 3$.
 - (c) $q = 2, n = 34, M = 37, d = 17$.
 - (d) $q = 41, n = 41, M = 41, d = 41$.
 - (e) $q = 4, n = 6, M = 65, d = 4$.
 - (f) $q = 2, n = 12, M = 16, d = 5$.
 - (g) $q = 2, n = 8, M = 17, d = 4$.
 - (h) $q = 3, n = 6, M = 4, d = 5$.

Model solutions and mark scheme

B = bookwork, C= similar to coursework, U = unseen.

1. (a) (B) Given a word $v \in \mathcal{C}$, let v' be the word of length $n - 1$ obtained by deleting the last symbol from v . Let $\mathcal{C}' = \{v' \mid v \in \mathcal{C}\}$.

Given distinct words $v, w \in \mathcal{C}$, we have $d(v, w) \geq d$, so v and w differ in at least d positions. At most one of these positions can be the last one, so v' and w' differ in at least $d - 1$ positions, i.e. $d(v', w') \geq d - 1$. In particular, since $d - 1 \geq 1$, v' and w' are distinct, so there are M distinct words in \mathcal{C}' . So \mathcal{C}' is an $(n - 1, M, d - 1)$ -code.

We can find an (n, M, d) -code with $M = A_q(n, d)$, and the construction then shows that an $(n - 1, M, d - 1)$ -code exists, i.e. $A_q(n - 1, d - 1) \geq M$. So $A_q(n - 1, d - 1) \geq A_q(n, d)$. (Marks: 8. 3 for defining the code \mathcal{C}' accurately, 5 for the proof.)

- (b) (B) For each word v in \mathcal{D} , we form the word \hat{v} of length n by adding a 0 or a 1 in such a way as to make the total number of 1s in \hat{v} even. We let $\hat{\mathcal{D}} = \{\hat{v} \mid v \in \mathcal{D}\}$. If v, w are distinct words in \mathcal{D} , then we have $d(v, w) \geq d - 1$, so v and w differ in at least $d - 1$ positions. So \hat{v} and \hat{w} differ in at least $d - 1$ positions, so $d(\hat{v}, \hat{w}) \geq d - 1$. In particular, \hat{v} and \hat{w} are distinct, so there are M words in $\hat{\mathcal{D}}$. Now we claim that $d(\hat{v}, \hat{w})$ is even for all $v, w \in \mathcal{D}$, which will mean that $d(\hat{v}, \hat{w}) \geq d$ if v and w are distinct. To prove the claim, we notice that

$$\begin{aligned} d(\hat{v}, \hat{w}) &= (\text{no. of places where } \hat{v} \text{ and } \hat{w} \text{ differ}) \\ &= (\text{no. of places where } \hat{v} \text{ has a 0 and } \hat{w} \text{ has a 1}) \\ &\quad + (\text{no. of places where } \hat{v} \text{ has a 1 and } \hat{w} \text{ has a 0}) \\ &= (\text{no. of places where } \hat{v} \text{ has a 0 and } \hat{w} \text{ has a 1}) \\ &\quad + (\text{no. of places where } \hat{v} \text{ has a 1 and } \hat{w} \text{ has a 1}) \\ &\quad + (\text{no. of places where } \hat{v} \text{ has a 1 and } \hat{w} \text{ has a 0}) \\ &\quad + (\text{no. of places where } \hat{v} \text{ has a 1 and } \hat{w} \text{ has a 1}) \\ &\quad - 2(\text{no. of places where } \hat{v} \text{ has a 1 and } \hat{w} \text{ has a 1}) \\ &= (\text{no. of places where } \hat{v} \text{ has a 1}) \\ &\quad + (\text{no. of places where } \hat{w} \text{ has a 1}) \\ &\quad - 2(\text{no. of places where } \hat{v} \text{ has a 1 and } \hat{w} \text{ has a 1}). \end{aligned}$$

This is the sum of three even numbers, so is even. Hence $d(\hat{v}, \hat{w}) \geq d$ whenever v and w are distinct, so $\hat{\mathcal{D}}$ is an (n, M, d) -code.

(Marks: 9. 3 for defining the code $\hat{\mathcal{C}}$ accurately, 3 for proving that $d(\hat{v}, \hat{w})$ is even, 3 for finishing the proof.)

- (c) (B) We have $A_2(n, d) \leq A_2(n - 1, d - 1)$ from (b). In (c), if we start with an $(n - 1, M, d - 1)$ -code for $M = A_2(n - 1, d - 1)$, we obtain an (n, M, d) -code, and this implies that $A_2(n, d) \geq M$, i.e. $A_2(n, d) \geq A_2(n - 1, d - 1)$. So we have $A_2(n, d) = A_2(n - 1, d - 1)$ if d is even.

(Marks: 2.)

- (d) (U) $A_2(3, 2) \geq 4$, since the following is a binary $(3, 4, 2)$ -code: $\{000, 011, 101, 110\}$. By the Plotkin bound,

$$A_2(4, 3) \leq 2 \left\lfloor \frac{4}{3} \right\rfloor = 2.$$

So $A_2(3, 2) > A_2(4, 3)$.

(Marks: 6. 3 for showing a lower bound on $A_2(n-1, d-1)$ and the other 3 for an upper bound on $A_2(n, d)$. An accurate statement of the Plotkin bound is not required, as long as it's clear where they're using it and they actually get the correct bound given by Plotkin (e.g. if in the above example they say 'by Plotkin, $A_2(4, 3) \leq 3$ ', they don't get the marks.)

2. (a) (B) A linear $[n, k]$ -code is a subspace \mathcal{C} of the vector space \mathbb{F}_q^n of dimension k .
(Marks: 1.)
- (b) (B) If \mathcal{C} is a linear $[n, k]$ -code, a generator matrix for \mathcal{C} is a $k \times n$ matrix G such that the rows of G form a basis for \mathcal{C} .
(Marks: 1.)
- (c) (B) We define the function $\cdot : \mathbb{F}_q^n \times \mathbb{F}_q^n \rightarrow \mathbb{F}_q$ by

$$v \cdot w = \sum_{i=1}^n v_i w_i.$$

Then we define $\mathcal{C}^\perp = \{v \in \mathbb{F}_q^n \mid v \cdot w = 0 \text{ for all } w \in \mathcal{C}\}$.

(Marks: 2. 1 for defining the dot product, 1 for the rest.)

- (d) (B) v lies in \mathcal{C}^\perp if and only if $Gv^T = 0$.
(Marks: 1.)
- (e) (B) The above criterion says that \mathcal{C}^\perp is the kernel of G . The rank-nullity theorem says that the kernel of G is a subspace of \mathbb{F}_q^n of dimension n minus the rank of G . The rank of G is the maximum number of linearly independent rows of G . The rows of G form a basis of \mathcal{C} , so are linearly independent. So the rank of G is the number of rows, i.e. k . So \mathcal{C}^\perp is a subspace of \mathbb{F}_q^n of dimension $n - k$, i.e. a linear $[n, n - k]$ -code.
(Marks: 5. 3 for applying the rank-nullity theorem, 2 for calculating the rank of G . An accurate statement of the rank-nullity theorem is not required, as long as the application is clear.)
- (f) (B) By the previous part, \mathcal{C}^\perp is a linear $[n, n - k]$ -code, so $(\mathcal{C}^\perp)^\perp$ is a linear $[n, n - (n - k)]$ -code, i.e. a linear $[n, k]$ -code. Now we claim that $\mathcal{C} \subseteq (\mathcal{C}^\perp)^\perp$: since \mathcal{C}^\perp is the set of words v such that $v \cdot w = 0$ for all $w \in \mathcal{C}$, we have $v \cdot w = 0$ for all $w \in \mathcal{C}$ and all $v \in \mathcal{C}^\perp$. Hence if $w \in \mathcal{C}$, then $v \cdot w = 0$ for all $v \in \mathcal{C}^\perp$, which means that $w \in (\mathcal{C}^\perp)^\perp$. So $\mathcal{C} \subseteq (\mathcal{C}^\perp)^\perp$. If $\mathcal{C} \subset (\mathcal{C}^\perp)^\perp$ then we would have $\dim \mathcal{C} < \dim(\mathcal{C}^\perp)^\perp$. But we have seen that $\dim \mathcal{C} = \dim(\mathcal{C}^\perp)^\perp$, and so $\mathcal{C} = (\mathcal{C}^\perp)^\perp$.
(Marks: 5. 3 for showing that $\mathcal{C} \subseteq (\mathcal{C}^\perp)^\perp$, 2 for applying the previous part twice.)

- (g) (B) A generator matrix is in standard form if it is of the form $(I_k|A)$, where I_k is the $k \times k$ identity matrix and A is some $k \times n - k$ matrix. A parity-check matrix for \mathcal{C} is a generator matrix for \mathcal{C}^\perp . If $G = (I_k|A)$ is a generator matrix for \mathcal{C} , then the matrix $H = (-A^T|I_{n-k})$ is a generator matrix for \mathcal{C} .

(Marks: 4. 1 for defining a parity-check matrix, 1 for standard form, 2 for finding a parity-check matrix from a standard-form generator matrix.)

- (h) (U) $\mathcal{C} = \{00000, 01101, 10110, 11011\}$ is a binary $[5, 2, 3]$ -code. This has a basis $\{10110, 01101\}$, and so it has a standard-form generator matrix

$$\begin{pmatrix} 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{pmatrix}$$

and hence a parity-check matrix

$$\begin{pmatrix} 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{pmatrix}.$$

$\text{Ham}(2, 5)$ has a parity-check matrix whose columns are the non-zero vectors in \mathbb{F}_5^2 , up to scaling, e.g.

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 2 & 3 & 4 & 0 & 1 \end{pmatrix}.$$

Hence it has a generator matrix

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 4 & 4 \\ 0 & 1 & 0 & 0 & 4 & 3 \\ 0 & 0 & 1 & 0 & 4 & 2 \\ 0 & 0 & 0 & 1 & 4 & 1 \end{pmatrix}.$$

Marks: 6. None for finding a $[5, 2, 3]$ -code, since it's written elsewhere in the paper. $1\frac{1}{2}$ for each matrix, dropping $\frac{1}{2}$ if the method is clear but arithmetic errors are made. If the first matrix for either code is wrong, they can still get the marks for the second if it's correct modulo the error.)

3. (a) (B) If A is an alphabet and \mathcal{C} is a code of length n over A , then \mathcal{C} is t -error-correcting if there do not exist $v \neq w \in \mathcal{C}$ and $x \in A^n$ such that $d(v, x) \leq t$ and $d(w, x) \leq t$.

Suppose $d(\mathcal{C}) \leq 2t$, so that there exist distinct words $v, w \in \mathcal{C}$ with $d(v, w) \leq 2t$. Then we can change v into w by changing at most $2t$ symbols. Let x be a word obtained by making only t of these changes. Then x differs from v in exactly t positions, and x differs from w in at most t positions, since we can get from x to w by changing at most t more symbols. So we have $v \neq w \in \mathcal{C}$ and $x \in A^n$ such that $d(v, x) \leq t$ and $d(w, x) \leq t$, so \mathcal{C} is not t -error-correcting.

Conversely, suppose \mathcal{C} is not t -error-correcting, and take $v \neq w \in \mathcal{C}$ and $x \in A^n$ such that $d(v, x) \leq t$ and $d(w, x) \leq t$. Then v differs from x in at most t positions, and x differs from w in at most t positions, and so v agrees with w everywhere except at most $2t$ positions, so $d(v, w) \leq 2t$. So the minimum distance of \mathcal{C} is less than $2t + 1$.

(Marks: 8. 2 for a correct definition, 3 for each direction of the proof.)

- (b) (B) Hamming bound: if A is a q -ary alphabet, and \mathcal{C} is a t -error-correcting code in A^n , then

$$|\mathcal{C}| \leq \frac{q^n}{\binom{n}{0} + (q-1)\binom{n}{1} + (q-1)^2\binom{n}{2} + \cdots + (q-1)^t\binom{n}{t}}.$$

If \mathcal{C} is a t -error-correcting code in A^n , then \mathcal{C} is perfect if

$$|\mathcal{C}| = \frac{q^n}{\binom{n}{0} + (q-1)\binom{n}{1} + (q-1)^2\binom{n}{2} + \cdots + (q-1)^t\binom{n}{t}}.$$

(Marks: 3. 2 for the bound, 1 for the definition of perfect.)

- (c) (B) **Basic fact.** The minimum distance of a linear code equals the minimum weight of a non-zero codeword.

Basic fact. If G is a $k \times n$ matrix of rank k , then the set of words v such that $Gv^T = 0$ is a linear code of dimension $n - k$.

Given $r > 0$, we construct an $r \times (2^r - 1)$ matrix G whose columns are all possible non-zero vectors in \mathbb{F}_2^r , in any order. We then define $\text{Ham}(r, 2)$ to be the linear code with G as parity-check matrix, i.e. $\text{Ham}(r, 2)$ is the set of all words $w \in \mathbb{F}_2^{2^r - 1}$ such that $Gw^T = 0$.

G certainly has r linearly independent columns, namely the standard basis vectors of \mathbb{F}_2^r , so the rank of G is r . So $\text{Ham}(r, 2)$ is a linear code of dimension $2^r - 1 - r$ over \mathbb{F}_2 . We claim that the minimum distance of $\text{Ham}(r, 2)$ is at least 3. This holds if and only if every non-zero word in $\text{Ham}(r, 2)$ has weight at least 3.

Suppose w is a word of weight 1, with a 1 in position i and 0s elsewhere. Then Gw^T is equal to the i th column of G , and this is non-zero, so w does not lie in $\text{Ham}(r, 2)$.

Now suppose w is a word of weight 2, with 1s in positions i and j and 0s elsewhere. Then Gw^T is equal to the i th column of G plus the j th column of G . Since we are working in binary, this is the same as the i th column of G minus the j th column of G , and this is not zero, since the columns are distinct. So w does not lie in $\text{Ham}(r, 2)$.

So there are no words of weight 1 or 2 in $\text{Ham}(r, 2)$, and so $\text{Ham}(r, 2)$ has minimum distance at least 3, and so $\text{Ham}(r, 2)$ is 1-error-correcting.

We have $|\text{Ham}(r, 2)| = 2^{\dim(\text{Ham}(r, 2))} = 2^{N-r}$, where $N = 2^r - 1$. So $\text{Ham}(r, 2)$ is a $(N, 2^{N-r}, 3)$ -code. We have

$$\frac{2^N}{\binom{N}{0} + (2-1)^1\binom{N}{1}} = \frac{2^N}{1+N} = \frac{2^N}{2^r} = 2^{N-r} = |\text{Ham}(r, 2)|,$$

and so $\text{Ham}(2, r)$ is perfect.

For $\text{Ham}(3, 2)$, we can take as a parity-check matrix

$$\begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 \end{pmatrix},$$

and we get a generator matrix

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix}.$$

(Marks: 14. 4 for accurate construction of $\text{Ham}(r, 2)$, 4 for evaluating minimum distance, 4 for checking the Hamming bound, 1 for each matrix.)

4. (a) (B) If \mathcal{C} is a linear code of length n over \mathbb{F}_q , we define the following two operations:

Operation 1 Choose a permutation σ of $\{1, \dots, n\}$, and for a word $v = v_1 \dots v_n \in \mathbb{F}_q^n$ define

$$v_\sigma = v_{\sigma(1)} \dots v_{\sigma(n)}.$$

Now replace \mathcal{C} with the code

$$\mathcal{C}_\sigma = \{v_\sigma \mid v \in \mathcal{C}\}.$$

Operation 2' Choose a non-zero element a of \mathbb{F}_q and an integer $i \in \{1, \dots, n\}$. For a word $v = v_1 \dots v_n \in \mathbb{F}_q^n$ define

$$v_{a,i} = v_1 \dots v_{i-1} (av_i) v_{i+1} \dots v_n.$$

Now replace \mathcal{C} with the code

$$\mathcal{C}_{a,i} = \{v_{a,i} \mid v \in \mathcal{C}\}.$$

We say that \mathcal{C} and \mathcal{D} are equivalent if we can get from \mathcal{C} to \mathcal{D} by applying a sequence of operations of these two types. (Marks: 4. 3 for describing the operations, 1 for the rest.)

- (b) (C) For Operation 1, we claim that $\text{weight}(v_\sigma) = \text{weight}(v)$ for all v : this is because v_σ and v have the same symbols, but in a different order, and so they both have the same number of non-zero symbols. Since $v \mapsto v_\sigma$ is a bijection, the number of words in \mathcal{C}_σ of weight w will equal the number of words in \mathcal{C} of weight w .

For Operation 2', we claim that $\text{weight}(v_{a,i}) = \text{weight}(v)$ for all v : we have av_i non-zero if and only if v_i is non-zero, so in fact for each j the j th symbol of $v_{a,i}$ will be non-zero if and only if the j th symbol of v is non-zero. So both words have the same number of non-zero symbols. Since $v \mapsto v_{a,i}$ is a bijection, the number of words in $\mathcal{C}_{a,i}$ of weight w will equal the number of words in \mathcal{C} of weight w .

So Operations 1 and 2' both preserve the number of words of weight w , so any two equivalent codes will have the same number of words of weight w .

(Marks: 6. 2 for Operation 1, 3 for Operation 2', 1 for finishing the proof.)

- (c) (B) We may use the following five matrix operations:

MO1 Permute the rows of a matrix.

MO2 Multiply a row of a matrix by a non-zero element of \mathbb{F}_q .

MO3 Add a multiple of one row of a matrix to another row of a matrix.

MO4 Permute the columns of a matrix.

MO5 Multiply a column of a matrix by a non-zero element of \mathbb{F}_q .

If we start with a generator matrix for \mathcal{C} and apply any of these operations, then we get a generator matrix for a code equivalent to \mathcal{C} .

(Marks: 3. 1 mark for each operation, but only the worst three count. I'm happy with reasonably informal definitions like I've given above, but it's crucial to say 'non-zero' in MO2 and MO5.)

- (d) (B/C) **Fact.** If G is a generator matrix for a code \mathcal{C} , then by applying matrix operations MO1–MO5, we may put G in *standard form*, i.e. we may transform G into a matrix $(I|A)$, where I is an identity matrix.

If \mathcal{C} is a $[3, 2]$ -code, then \mathcal{C} has a generator matrix G which is a 2×3 matrix over \mathbb{F}_q . By the above fact, we may put G in standard form, i.e. we may replace G with

$$\begin{pmatrix} 1 & 0 & a \\ 0 & 1 & b \end{pmatrix}$$

for some $a, b \in \mathbb{F}_q$.

If $a = b = 0$, then this is the third matrix given. If $a = 0$ and $b \neq 0$, then we may apply MO5, multiplying column 3 by b^{-1} , and get the matrix

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \end{pmatrix}.$$

If $a \neq 0$, then we apply MO5, multiplying column 3 by a^{-1} to get the matrix

$$\begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & c \end{pmatrix}$$

for some c . We now consider separately the cases $c = 0$ and $c \neq 0$.

If $c = 0$, then we apply MO1, swapping rows 1 and 2, and then we apply MO4, swapping columns 1 and 2, to get the matrix

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \end{pmatrix}.$$

If $c \neq 0$, then we apply MO2, multiplying row 2 by c^{-1} , and then we apply MO5, multiplying column 2 by c to get the matrix

$$\begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix}.$$

Hence we can always transform G into one of the given matrices.

(Marks: 7. 2 for accurately stating the standard form result, 5 for checking through the possibilities.)

- (e) (U) **Fact.** The minimum distance of a linear code equals the minimum weight of a non-zero codeword.

By the previous parts, we find that any $[3, 2]$ -code \mathcal{C} is equivalent to one of the three codes $\mathcal{D}, \mathcal{E}, \mathcal{F}$ with generator matrices

$$\begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix}, \quad \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \end{pmatrix}, \quad \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$$

respectively. \mathcal{E} and \mathcal{F} both contain the word 001, which has weight 1. If \mathcal{C} has minimum distance at least 2, then \mathcal{C} cannot contain a word of weight 1, and so by part (b) \mathcal{C} cannot be equivalent to \mathcal{E} or \mathcal{F} . So \mathcal{C} is equivalent to \mathcal{D} .

(Marks: 5. 1 for writing down the three codes, 2 for observing that two of them minimum distance 1, 2 for applying part (b).)

5. (a) (B) If \mathcal{C} is a linear $[n, k]$ -code over \mathbb{F}_q , a coset of \mathcal{C} is a set of the form

$$w + \mathcal{C} = \{w + v \mid v \in \mathcal{C}\}$$

for $w \in \mathbb{F}_q^n$. A leader for a coset is an element of minimal weight in that coset. To construct a Slepian array:

- choose a leader in each coset;
- in the first row of the array, write the codewords, with the word $00 \dots 0$ at the left, and the remaining codewords in any order;
- in the first column write the chosen coset leaders, with $00 \dots 0$ of the coset \mathcal{C} at the top, and the remaining coset leaders in any order;
- for the entry in row i and column j , we put the word which equals the coset leader at the start of row i plus the codeword at the top of column j .

(Marks: 5. 1 for defining a coset, 1 for a leader, 3 for a Slepian array.)

- (b) (B) Suppose w is a word in \mathbb{F}_q^n . Then $w = w + 00 \dots 0$ lies in the coset $w + \mathcal{C}$. Let v be the chosen leader for this coset. Then $v = w + x$, for some codeword x . Since \mathcal{C} is linear, the word $-x$ is also a codeword, and so $w = v + (-x)$ lies in array in the row with v at the beginning, and in the column with $-x$ at the top.

We form a decoding process $f : \mathbb{F}_q^n \rightarrow \mathcal{C}$ as follows: given a word $w \in \mathbb{F}_q^n$, find w in the Slepian array. Then set $f(w)$ to be the codeword at the top of the same column.

(Marks: 4. 2 for the proof, 2 for the definition of the decoding process.)

- (c) (C)

00000	01101	10110	11011
00001	01100	10111	11010
00010	01111	10100	11001
00100	01001	10010	11111
01000	00101	11110	10011
10000	11101	00110	01011
00011	01110	10101	11000
01010	00111	11100	10001

(Marks: 4.)

(d) (B) Define the scalar product $\cdot : \mathbb{F}_q^n \times \mathbb{F}_q^n \rightarrow \mathbb{F}_q$ by

$$v \cdot w = \sum_{i=1}^n v_i w_i.$$

Define $\mathcal{C}^\perp = \{w \in \mathbb{F}_q^n \mid v \cdot w = 0 \text{ for all } v \in \mathcal{C}\}$. A parity-check matrix for \mathcal{C} is a generator matrix for \mathcal{C}^\perp , that is, a matrix whose rows form a basis of \mathcal{C}^\perp . If H is a parity-check matrix for \mathcal{C} and w is a word in \mathbb{F}_q^n , the syndrome of w is the word wH^T . A syndrome look-up table has two columns; in the first column there are coset leaders, one from each coset. In the second column, next to the coset leader w we write the syndrome of w . Given a syndrome look-up table, we form a decoding process $f : \mathbb{F}_q^n \rightarrow \mathcal{C}$ as follows. Given a word $w \in \mathbb{F}_q^n$, calculate its syndrome. Find this syndrome in the look-up table, and let v be the coset leader with the same syndrome. Set $f(w) = w - v$.

(Marks: 7. 2 for a parity-check matrix, 1 for a syndrome, 2 for a syndrome look-up table, 2 for the decoding process.)

(e) (U) $\{10101, 01010\}$ is a basis for \mathcal{D} , so \mathcal{D} has a generator matrix

$$\begin{pmatrix} 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \end{pmatrix}.$$

\mathcal{D} has a parity-check matrix

$$\begin{pmatrix} 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

\mathcal{D} has a syndrome look-up table

00000	000
00001	001
00010	010
00100	100
10000	101
00011	011
00110	110
11000	111

(Marks: 5. 2 for the parity-check matrix, 3 for the syndrome table.)

6. (a) (U) We find the code generated by the given matrix:

$$\{000000, 110001, 101010, 011100, 011011, 101101, 110110, 000111\}.$$

Now we add a parity-check digit to the end of each word:

$$\{0000000, 1100011, 1010101, 0111001, 0110110, 1011010, 1101100, 0001111\}.$$

By inspection, this is a $(7, 8, 4)$ -code.

(Marks: 5. These can be obtained for a $(7, 8, 4)$ -code with minimal explanation.)

- (b) (C) For each word $w \in \mathcal{C}$, we form the word w^+ of length nm by writing w m times. We let $\mathcal{C}^+ = \{w^+ \mid w \in \mathcal{C}\}$. Clearly if v, w are distinct words in \mathcal{C} , then v^+ and w^+ are distinct, so there are M words in \mathcal{C}^+ . It remains to show that \mathcal{C}^+ had minimum distance at least dm . If x, y are distinct words in \mathcal{C}^+ , then $x = v^+, y = w^+$ for distinct words $v, w \in \mathcal{C}$. We have $d(v, w) \geq d$, so v and w differ in at least d positions. So v^+ and w^+ differ in at least d of the first n positions, and at least d of the next n positions, and at least d of the next n , and so on, and so they differ in at least dm positions. So $d(v^+, w^+) \geq md$.

(Marks: 7. 2 for the construction of the code, 5 for showing that it works.)

- (c) (B) The Plotkin bound: suppose n, d are positive integers.
- If d is even and $2d > n$, then

$$A_2(n, d) \leq 2 \left\lfloor \frac{d}{2d - n} \right\rfloor.$$

- If d is odd and $2d + 1 > n$, then

$$A_2(n, d) \leq 2 \left\lfloor \frac{d + 1}{2d + 1 - n} \right\rfloor.$$

(Marks: 3.)

- (d) (U/C) $4m$ is even and $2 \times 4m > 7m$, so by the Plotkin bound we have

$$A_2(7m, 4m) \leq 2 \left\lfloor \frac{4m}{m} \right\rfloor = 8.$$

The first part of this question gives a binary $(7, 8, 4)$ -code, and using this and the second part of the question, we get a binary $(7m, 8, 4m)$ -code for any m . Hence $A_2(7m, 4m) \geq 8$, and we're done.

(Marks: 5. 2 for applying Plotkin, 3 for getting the lower bound.)

- (e) (U/B) The Plotkin bound gives

$$A_2(7m - 1, 4m) \leq 2 \left\lfloor \frac{4m}{m + 1} \right\rfloor;$$

we have $4m/(m + 1) < 4$, so $A_2(7m - 1, 4m) \leq 6$.

The $(7m, 8, 4m)$ -code constructed above has four words ending in 0. Any two of these words differ in at least $4m$ positions, and none of these positions is the last position. So if we take the four words ending in 0 and delete the last symbol from each, we shall have a $(7m - 1, 4, 4m)$ -code. So $A_2(7m - 1, 4m) \geq 4$.

(Marks: 5. 1 for applying Plotkin, 4 for the rest.)

7. (a) (B) If $\mathcal{C} \subseteq A^n$, a nearest-neighbour decoding process is a function $f : A^n \rightarrow \mathcal{C}$ such that

$$d(v, f(v)) \leq d(v, w)$$

for all $v \in A^n$ and $w \in \mathcal{C}$.

(Marks: 3.)

(b)

0000 \mapsto 0000
0001 \mapsto 0000
0010 \mapsto 0000
0011 \mapsto 0111
0100 \mapsto 0000
0101 \mapsto 0111
0110 \mapsto 0111
0111 \mapsto 0111
1000 \mapsto 0000
1001 \mapsto 1011
1010 \mapsto 1110
1011 \mapsto 1011
1100 \mapsto 1110
1101 \mapsto 1101
1110 \mapsto 1110
1111 \mapsto 0111

(Marks: 6. Lose $\frac{1}{2}$ for each error, down to a minimum of 0.)

(c) The error probability is the probability that we get one of the words

0000,0001,0010,0100,1000,1001,1010,1011,1100,1101,1110,

i.e.

$$\frac{3}{4^4} + \frac{3^2}{4^4} + \frac{3^2}{4^4} + \frac{3^2}{4^4} + \frac{1}{4^4} + \frac{3}{4^4} + \frac{3}{4^4} + \frac{3^2}{4^4} + \frac{3}{4^4} + \frac{3^2}{4^4} + \frac{3^2}{4^4} = \frac{67}{256}.$$

(Marks: 5. 3 of these are available for making it clear that they know what the word error probability is, and what they're calculating.)

(d) (B) The rate of a binary (n, M, d) -code is $\log_2 M/n$. The capacity of a binary symmetric channel with symbol error probability is $1 + p \log_2 p + (1 - p) \log_2(1 - p)$.

Shannon's Theorem: Suppose we have a binary symmetric channel with capacity C . Suppose ϵ and ρ are positive real numbers with $\rho < C$. Then for any sufficiently large n , there exists a binary code of length n and rate at least ρ and a decoding procedure such that the word error probability is at most ϵ .

(Marks: 5. 1 for rate, 1 for capacity, 3 for Shannon.)

(e) (U) The binary repetition code is the code over $\{0, 1\}$ with two words $00 \dots 0$ and $11 \dots 1$. It has a unique nearest-neighbour decoding process if for every word w of length n over $0, 1$, there is a unique nearest codeword to w , i.e. $d(00 \dots 0, w) \neq d(11 \dots 1, w)$. If w

contains x 0s and $n-x$ 1s, then we have $d(00 \dots 0, w) = n-x$ and $d(11 \dots 1, w) = x$. If n is even, then we can take a word w with $x = n/2$, and we get $d(00 \dots 0, w) = d(11 \dots 1, w)$, so the repetition code does not have a unique nearest-neighbour decoding process. If n is odd, then we cannot have $x = n-x$ for any x , so the repetition code does have a unique nearest-neighbour decoding process.

(Marks: 6.)

8. We quote the following.

Plotkin bound: if n, d are positive integers with d odd and $2d+1 > n$, then

$$A_2(n, d) \leq 2 \left\lfloor \frac{d+1}{2d+1-n} \right\rfloor.$$

Hamming bound: if \mathcal{C} is a q -ary (n, M, d) -code with $d > 2t$, then

$$M \leq \frac{q^n}{\binom{n}{0} + (q-1)\binom{n}{1} + \dots + (q-1)^t \binom{n}{t}}.$$

Singleton bound:

(a) If $n, d > 1$ then

$$A_q(n, d) \leq A_q(n-1, d-1);$$

(b) If $n, d \geq 1$ then

$$A_q(n, d) \leq q^{n-d+1}.$$

Gilbert–Varshamov bound: If q is a prime power and n, r, d are positive integers satisfying

$$\binom{n-1}{0} + (q-1)\binom{n-1}{1} + \dots + (q-1)^{d-2}\binom{n-1}{d-2} < q^r,$$

then a linear $[n, n-r, d]$ -code over \mathbb{F}_q exists.

(a) (U) No. We apply the Hamming bound: if \mathcal{C} is a binary $(12, M, 5)$ -code, then

$$M \leq \frac{2^{12}}{\binom{12}{0} + \binom{12}{1} + \binom{12}{2}} = \frac{4096}{79} < 52.$$

(Marks: 3.)

(b) (U) Yes. We take the Hamming code $\text{Ham}(2, 3)$:

$$\{0000, 1011, 0112, 2022, 0221, 1120, 2210, 1202, 2101\}.$$

(Marks: 3.)

(c) (U) No. We apply the Plotkin bound:

$$A_2(34, 17) \leq 2 \left\lfloor \frac{18}{1} \right\rfloor = 36.$$

(Marks: 3.)

(d) (U) Yes. We take the alphabet $\{0, 1, \dots, 40\}$ and construct the repetition code

$$\{00 \dots 0, 11 \dots 1, 22 \dots 2, \dots, (40)(40) \dots (40)\}.$$

(Marks: 3.)

(e) (U) No. We apply the Singleton bound (b):

$$A_4(6, 4) \leq 4^{6-4+1} = 64.$$

(Marks: 3.)

(f) (U) Yes. We use the Gilbert–Varshamov bound, putting $q = 2$, $n = 12$, $r = 8$, $d = 5$.

We have

$$\binom{11}{0} + \binom{11}{1} + \binom{11}{2} + \binom{11}{3} = 232 < 2^8.$$

So a binary $[12, 4, 5]$ -code exists, and this is a binary $(12, 16, 5)$ -code.

(Marks: 3.)

(g) (U) No. By part (a) of the Singleton bound, we have

$$A_2(8, 4) \leq A_2(7, 3).$$

Now we apply the Hamming bound:

$$A_2(7, 3) \leq \frac{2^7}{\binom{7}{0} + \binom{7}{1}} = 16.$$

(Marks: 4. 2 for each part.)

(h) (C) Yes:

$$\{000000, 011111, 222210, 101222\}.$$

(Marks: 3.)

(All parts lose marks for inaccurate statements of the bounds they use.)