4F5 Advanced Information Theory and Coding 2025 Crib

Question 1

(a) (i) $2^6 = 64$ and hence $R_9(2^6) = 1$. Furthermore, $R_9(2^{6k}) = R_9((2^6)^k) = 1$.

(ii) The equation on the right follows directly from (i). To establish the equation on the left, note that if $R_a(b) = c$ where a is odd, and b and c are even, it must hold that for some quotient q, $qa + c = b$. Since b and c are even, qa and hence q must be even, implying that $\frac{q}{2} \cdot 2a + c = b$ and hence $R_{2a}(b) = c$. In the equation, 2^{6k+2} and 4 are even and 9 is odd, showing that $R_9(2^{6k+2}) = R_{2\times 9}(2^{6k+2}) = R_{18}(2^{6k+2}) = 4$.

(iii) The multiplicative group of $GF(19)$ has order 18 and hence every element in the group has multiplicative order dividing 18, and hence $R_{19}(a^{18}) = 1$ for any a between 1 and 18, including 2.

(iv) (ii) implies that there exists a quotient q such that $2^{6k+2} = 18q + 4$ which explains the inequality on the left. Furthermore, $R_{19}(2^{18q+4}) = R_{19}(2^4)$ follows from (iii).

The proof is completed by noting that if $R_{19}(2^{2^{6k+2}}) = 16$ then $R_{19}(3 + 2^{2^{6k+2}}) = 0$.

(b) (i) This is just binary elementwise addition, i.e. $(1 + X + X^3) + (X + X^2 + X^3) = 1 + X^2$.

(ii) We can pre-compute the products of $X + X^2 + X^3$ with elements X^k for $k = 0, 1, 2, 3, 4$,

$$\begin{cases} X^0(X + X^2 + X^3) & = X + X^2 + X^3 \\ X^1(X + X^2 + X^3) & = X^2 + X^3 + X^4 \\ X^2(X + X^2 + X^3) & = 1 + X^2 + X^3 + X^4 \\ X^3(X + X^2 + X^3) & = 1 + X + X^2 + X^3 + X^4 \\ X^4(X + X^2 + X^3) & = 1 + X + X^3 + X^4 \end{cases}$$

Then compute $(1 + X + X^3)(X + X^2 + X^3)$ as the sum of $1(X + X^2 + X^3)$, $X(X + X^2 + X^3)$ and $X^3(X + X^2 + X^3)$,

$$(1 + X + X^3)(X + X^2 + X^3) = 1 + X^2 + X^3$$

(iii) Having pre-computed the products of $X+X^2+X^3$ in the previous question, we get the companion matrix simply by reading out the coefficients we computed

$$\boldsymbol{M} = \begin{bmatrix} 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 \end{bmatrix}$$

(iv) A primitive polynomial is an irreducible polynomial for which X generates the multiplicative group. Since the order of the multiplicative group of GF(32) is 31, a prime number, by Lagrange's theorem, all elements except 1 generate the group. Hence any irreducible binary polynomial of degree 5 is primitive and so $\pi(X)$ is primitive.

(v) For the same reason as in (iv), the multiplicative order of $1+X$ is 31.

(c) (i) This is easy to fix: Euler's function for the power of a prime p^e is not $(p-1)^2$ but $\varphi(p^e) = (p-1)p^{e-1}$. Hence, Bob should find a secret key d such that $R_w(d \times e) = 1$ for $w = (p-1)p = p^2 - p$.

(ii) $R_{p^2}((qp)^e) = R_{p^2}(R_{p2}(q^e)R_{p2}(p^e)) = R_{p^2}\left(R_{p2}(q^e)R_{p2}(p^2)R_{p^2}(p^{e-2})\right) = 0$ assuming $e \geq 2$, because $R_{p^2}(p^2) = 0$. Such a message would encrypt to zero and be undecryptable. Assuming p is a very large prime number, this is not a significant weakness, because the probability that a user might pick a message x to encrypyt that happens to be a multiple of p is vanishingly small.

(iii) We have $m = p^2 = 121$. The public key is $(m, e) = (121, 3)$. We compute d using Euclid's Extended gcd Algorithm with $\varphi(m) = p(p-1) = 110$

n_1	n_2	q	r	(a_1, b_1)	(a_2, b_2)
3	110	36	2	$(1, 0)$	$(0, 1)$
3	2	1	1	$(1, 0)$	$(-36, 1)$
1	2	2	0	$(37, -1)$	$(-36, 1)$
1	0			$(37, -1)$	$(-110, 3)$

Hence, we found that $d = 37$ and verify that $R_{110}(ed) = 1$. We encrypt the secret message $y = R_m(x^e) = R_{121}(1000) = 32$. To decrypt, we need to compute $R_m(y^d) = R_{121}(32^{37})$. This is a fairly large power but can be computed easily using the squares method

32^1	32^2	32^4	32^8	32^{16}	32^{32}
32	56	111	100	78	34

Now we note that $R_{121}(32^{37)} = R_{121}(32^{32}32^4 32^1) = R_{121}(34 \times 111 \times 32) = 10 = x$ so decryption was successful.

(iv) Kerckhoff's principle dictates that a system's safety must rely solely on the secret key and that a cryptographer must be able to publish every detail of the cryptosystem without compromising the system's safety. If Bob publishes the details of his cryptosystem, attackers will know that his system can be cracked by recovering p from m by taking the square root. Unlike factoring products of large primes which is a known hard problem, taking the square root of a large square prime is a very easy problem to solve and can be done by a simple binary search in no time at all. Once an attacker knows p, they can compute d and decrypt all messages.

Question 2

(a) This RS code has 13 parity symbols, which is an odd number. It can correct up to 6 errors, which is the same number of errors that could be corrected if we had only 12 parity symbols. Hence, the code can be improved by setting $K = 243$ instead of 242, improving the rate from $R = 242/255$ to $R = 243/255$ while correcting exactly the same number of errors. This is not the case for erasure channels: a (255,242) code can recover from 13 erasures while a (255,243) code can recover from only 12 errasures, so the extra parity symbol is not wasted as would be the case for error correction.

(b) The multiplicative group of GF(31) has order 30, so the code length must divide 30 by Lagrange's theorem: 30, 15, 10, 6, 5, 3, or 2.

(c)

$$\boldsymbol{F} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 4 & 16 & 2 & 8 \\ 1 & 16 & 8 & 4 & 2 \\ 1 & 2 & 4 & 8 & 16 \\ 1 & 8 & 2 & 16 & 4 \end{bmatrix}$$

The code length is $N = 5$.

(d) $K = 3$ so the code rate is $R = K/N = 3/5$. The parity-check matrix consists of the first 2 rows of the DFT matrix, verifying that the corresponding positions in the spectrum of the codeword are zero:

$$\boldsymbol{H} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 4 & 16 & 2 & 8 \end{bmatrix}$$

(e) The received word corresponds to the vector $\boldsymbol{r} = [2, 6, 8, 0, 26]$ in GF(31). We compute the DFT of $\boldsymbol{r}$,

$$\boldsymbol{R} = \boldsymbol{rF} = [11, 21, 28, 28, 15]$$

The first two elements are not zero, confirming that the channel has made at least one error. We now determine the length 1 recurrence relation that recovers the error sequence, by computing $21/11 = 16$ (this can be done by running the extended Euclid algorithm with 31 and 11 to find the inverse 17 of 11, then multiplying 17 by 21 mod 31 to yield 16.) Hence, the error sequence in the frequency domain is

$$\boldsymbol{E} = [11, 21, 26, 13, 22].$$

We obtain the codeword in the frequency domain as

$$\boldsymbol{C} = \boldsymbol{R} - \boldsymbol{E} = [0, 0, 2, 15, 24]$$

and convert $[2, 15, 24]$ back to characters to yield the word **BOX.**

(f) We insert zeros for erasures and convert the received word to the vector $\boldsymbol{r} = [9, 0, 1, 0, 0]$. We convert this word to the frequency domain

$$\boldsymbol{R} = \boldsymbol{rF} = [10, 25, 17, 13, 11].$$

Again, the fact that the first two symbols aren't zeros confirms that at least one of the erasures we replaced by a zero wasn't in fact a zero. To recover the error sequence, we need to compute the recurrence relation in the D domain based on the positions 3 and 4 of the erasures (being careful to count positions from 0 to 4)

$$c(D) = (1 - \alpha^3 D)(1 - \alpha^4 D) = (1 - 2D)(1 - 8D) = 1 - 10D + 16D^2 = 1 - 10D - 15D^2$$

This gives the relation

$$E_K - 10E_{k-1} - 15E_{k-2} = 0$$

or, equivalently,

$$E_k = 10E_{k-1} + 15E_{k-2}.$$

We apply this to the error sequence starting with $[10, 25]$ to recover the error sequence in the frequency domain

$$\boldsymbol{E} = [10, 25, 28, 4, 26]$$

and recover the codeword in the frequency domain

$$\boldsymbol{C} = \boldsymbol{R} - \boldsymbol{E} = [0, 0, 20, 9, 16]$$

and convert $[20, 9, 16]$ to the word **TIP**.

Question 3

(a) (i) This is the optimal source code and the probability of error is

$$p_e = \mathbb{P}[P_{V^n}(V^n) < \gamma(M)] \tag{1}$$

$$= \sum_{v^n : P_{V^n}(v^n) < \gamma(M)} P_{V^n}(v^n) \tag{2}$$

where $\gamma(M)$ is the probability below which codewords are assigned to the error index.

(ii) If we replace a sequence in the sum in (12) by one such that $P_{V^n}(v^n) > \gamma(M)$, the error probability will be higher.

(iii) We have that

$$M = \sum_{v^n} \mathbb{1}\{P_{V^n}(v^n) \geq \gamma(M)\} \tag{3}$$

$$\leq \sum_{v^n} \frac{P_{V^n}(v^n)}{\gamma(M)} \tag{4}$$

$$= \frac{1}{\gamma(M)} \tag{5}$$

Using the implied bound $\gamma(M) \leq \frac{1}{M}$ we have that

$$p_e \leq \mathbb{P}\left[P_{V^n}(V^n) < \frac{1}{M}\right] \tag{6}$$

which gives the result after taking logarithms and dividing by n.

(iv) Assuming $P_{V^n(v^n)} = \prod_{i=1}^n P_V(v_i)$ (discrete memoryless source assumption) and choosing $M = e^{n(H(V)+\delta)}$ for $\delta > 0$ we have that

$$p_e \leq \mathbb{P}\left[\frac{1}{n}\sum_{i=1}^n \log\frac{1}{P_V(V_i)} - H(V) > \delta\right] \tag{7}$$

$$\leq \mathbb{P}\left[\left|\frac{1}{n}\sum_{i=1}^n \log\frac{1}{P_V(V_i)} - H(V)\right| > \delta\right] \tag{8}$$

which tends to zero by the weak law of large numbers. This proves that for rates $R > H(V)$ the error probability tends to zero.

(b) (i) Stein's lemma states that for a binary hypothesis testing problem with iid distributions P_1^n, P_2^n, if $\pi_{1|2} \leq \beta$, where $\beta \in (0, 1)$ then

$$\lim_{n\to\infty} -\frac{1}{n}\log \pi_{2|1} = D(P_2 \| P_1).$$

The result means that for a fixed bound on one of the error probabilities, the other decays exponentially with exponent given by $D(P_2\|P_1)$. The result is important as the bigger $D(P_2\|P_1)$ the smaller the error probability (and the easier is to tell which distribution generated the observation).

(ii) If $\pi_{1|2} \leq 0.001$ and $\pi_{2|1} \leq 10^{-40}$ we have that

$$e^{-nD(P_2\|P_1)} \leq 10^{-40} \tag{9}$$

which is equivalent to

$$-nD(P_2\|P_1) \leq \log 10^{-40} \tag{10}$$

or

$$n \geq \frac{40\log 10}{D(P_2\|P_1)} \tag{11}$$

(iii) The result is the same, as Stein's lemma holds for any $\beta \in (0,1)$.

Question 4

(a) The probability of correct decoding averaged over the codebook can be written as

$$\bar{p}_c = \underbrace{\mathbb{E}\left[\frac{W^n(Y^n|X^n(1))}{\sum_{m'=1}^{M} W^n(Y^n|X^n(m'))}\right]}_{\text{expectation over all codebook}} \tag{12}$$

$$= \underbrace{\sum_{x^n,y^n} Q(x^n)W(y^n|x^n)}_{\text{expectation over codeword 1 and } y^n} \cdot \underbrace{\mathbb{E}\left[\frac{W^n(y^n|x^n)}{W^n(y^n|x^n) + \sum_{m'\neq 1}^{M} W^n(y^n|X^n(m'))}\right]}_{\text{expectation over rest of codebook}} \tag{13}$$

(b) The above can also be written as

$$\bar{p}_c = \sum_{x^n,y^n} Q(x^n)W(y^n|x^n)\mathbb{E}\left[\frac{1}{1+\sum_{m'\neq 1}^{M}\frac{W^n(y^n|X^n(m'))}{W^n(y^n|x^n)}}\right] \tag{14}$$

The function $f(x) = \frac{1}{1+x}$ is convex, and therefore, by Jensen's inequality (as given in the hint) we have

$$\bar{p}_c \geq \sum_{x^n,y^n} Q(x^n)W(y^n|x^n)\frac{1}{1+\mathbb{E}\left[\sum_{m'\neq 1}^{M}\frac{W^n(y^n|X^n(m'))}{W^n(y^n|x^n)}\right]} \tag{15}$$

(c) Since codewords are independent

$$\bar{p}_c \geq \sum_{x^n,y^n} Q(x^n)W(y^n|x^n)\frac{1}{1+\sum_{m'\neq 1}^{M}\mathbb{E}\left[\frac{W^n(y^n|X^n(m'))}{W^n(y^n|x^n)}\right]} \tag{16}$$

$$\sum_{x^n,y^n} Q(x^n)W(y^n|x^n)\frac{1}{1+(M-1)\mathbb{E}\left[\frac{W^n(y^n|\bar{X}^n)}{W^n(y^n|x^n)}\right]} \tag{17}$$

where the last step follows because the codewords are also identically distributed and hence all expectations are the same.

(d) We know that $\bar{p}_e = 1 - \bar{p}_c$. Applying $1 - \frac{1}{1+z} \leq \min\{1, z\}$ to (6) we get the result.

(e) Using the suggested inequality $\min\{1, x\} \leq x^{\rho}$, $\rho \in [0, 1]$, we have that

$$\bar{p}_e \leq M^{\rho} \sum_{x^n, y^n} Q(x^n) W^n(y^n|x^n) \left(\frac{\sum_{\bar{x}^n} Q(\bar{x}^n) W^n(y^n|\bar{x}^n)}{W^n(y^n|x^n)} \right)^{\rho} \tag{18}$$

When $Q(x^n) = \prod_{i=1}^n Q(x_i)$, we have that

$$\begin{aligned} \bar{p}_e &\leq M^{\rho} \left(\sum_{x,y} Q(x) W(y|x) \left(\frac{\sum_{\bar{x}} Q(\bar{x}) W(y|\bar{x})}{W(y|x)} \right)^{\rho} \right)^n & (19) \\ &= e^{-n(\bar{E}_0(\rho) - \rho R)} & (20) \end{aligned}$$

where $\bar{E}_0(\rho)$ is defined in the question.

(f) The function $\bar{E}_0(\rho)$ is such that $\bar{E}_0(0) = 0$, increasing in ρ and concave (as per the assumption), and we have that

$$\left. \frac{d\bar{E}_0(\rho)}{d\rho} \right|_{\rho=0} = I(X; Y). \tag{21}$$

Thus, since the overall exponent is $\max_{\rho \in [0,1]} \bar{E}_0(\rho) - \rho R$, the exponent is positive for all rates $R < I(X; Y)$, achieving exponentially vanishing error probabilities. We achieve capacity by optimizing over the input distribution $Q(x)$.