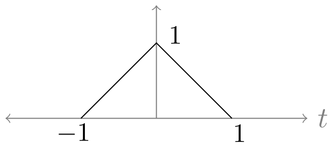
# 3F4 Data Transmission
# Engineering Tripos 2018/19 – Solutions

**Question** 1

(a) We have $g(t) = p(t) \star p(-t) = \int p(u)p(u-t)du$. Using this $g(t)$ is the triangular function below. [15%]




(b) From the above we note that $g(m-k) = 0$ for all integers $k \neq m$. Using this in the result of part (a), $r(m) = X_m + \tilde{n}(m)$. [5%]

(c) Sampling at time $(m - \Delta)$ for $m \geq 1$ gives the output [15%]

$$\begin{aligned} r(m-\Delta) &= \sum_{k=0}^{\infty} X_k g(m-k-\Delta) \ + \ \tilde{n}(m-\Delta) \\ &= X_m g(-\Delta) + X_{m-1} g(1-\Delta) + \tilde{n}(m-\Delta) \\ &= (1-\Delta)X_m \ + \ \Delta X_{m-1} \ + \ \tilde{n}(m-\Delta), \end{aligned} \tag{1}$$

where the values for $g(-\Delta)$ and $g(1-\Delta)$ are obtained from the sketch in part (b).

(d) If $0 < \Delta < \frac{1}{2}$, then $(1-\Delta) > \Delta$, so we detect $X_m$ from $r(m-\Delta)$ specified in (1). If $\frac{1}{2} < \Delta < 1$, then we detect symbol $X_m$ from $r(m+1-\Delta)$ which will contain a 'stronger' contribution from $X_m$: [15%]

$$r(m+1-\Delta) \ = \ \Delta X_m \ + \ (1-\Delta)X_{m+1} + \tilde{n}(m+1-\Delta). \tag{2}$$

(e) Since $\Delta = \frac{1}{2}$, we detect $X_m$ from (1). For brevity, we rewrite (1) as

$$r_m = (1-\Delta)X_m \ + \ \Delta X_{m-1} \ + \ \tilde{n}_m, \tag{3}$$

where $\tilde{n}_m \sim \mathcal{N}(0, \frac{N_0}{2})$. The detector declares $\hat{X}_m = A$ if $r_m > 0$, and $\hat{X}_m = -A$ otherwise. Due to symmetry of the constellation symbols, the probability of error can be expressed as [25%]

$$\begin{aligned} &P(\hat{X}_m \neq X_m) = P(\hat{X}_m \neq X_m \mid X_m = A) \\ &= \frac{1}{2}P(\hat{X}_m \neq X_m \mid X_m = A, \ X_{m-1} = A) + \frac{1}{2}P(\hat{X}_m \neq X_m \mid X_m = A, \ X_{m-1} = -A). \end{aligned} \tag{4}$$

The two terms are computed separately as follows:

$$\begin{aligned} P(\hat{X}_m \neq X_m \mid X_m = A, \ X_{m-1} = A) &= P((1-\Delta)A + \Delta A + \tilde{n}_m < 0 \mid X_m = A, \ X_{m-1} = A) \\ &= P(\tilde{n}_m < -A) = \mathcal{Q}\left(\frac{A}{\sqrt{N_0/2}}\right). \end{aligned} \tag{5}$$

Next,

$$P(\hat{X}_m \neq X_m \mid X_m = A,\ X_{m-1} = -A) = P((1-\Delta)A - \Delta A + \tilde{n}_m < 0 \mid X_m = A,\ X_{m-1} = A)$$
$$= P(\tilde{n}_m < -(1-2\Delta)A) = \mathcal{Q}\left(\frac{(1-2\Delta)A}{\sqrt{N_0/2}}\right). \tag{6}$$

Substituting (5) and (6) in (4), we obtain that the probability of detection error is

$$P(\hat{X}_m \neq X_m) = \frac{1}{2}\mathcal{Q}\left(\sqrt{\frac{2A^2}{N_0}}\right) + \frac{1}{2}\mathcal{Q}\left(\sqrt{\frac{2(1-2\Delta)^2A^2}{N_0}}\right).$$

(f) Let the coefficients of the 3-tap equaliser be $[h_0, h_1, h_2]$. With $\Delta = 0.1$, Eq. (3) becomes

$$r_m = 0.9X_m \ + \ 0.1X_{m-1} \ + \ \tilde{n}_m. \tag{7}$$

Letting $g_0 = 0.9$, $g_1 = 0.1$, the output of the FIR filter is [25%]

$$y_m = X_m f_0 \ + \sum_{j=1}^{3} X_{m-j} f_j + \sum_{i=0}^{2} h_i \tilde{n}_{m-i}, \tag{8}$$

where

$$\begin{aligned} f_0 &= h_0 g_0 = h_0(0.9), \\ f_1 &= h_0 g_1 + g_0 h_1 = h_0(0.1) + h_1(0.9), \\ f_2 &= h_1 g_1 + h_2 g_0 = h_1(0.1) + h_2(0.9), \\ f_3 &= h_2 g_1 = h_2(0.1). \end{aligned}$$

We set $f_0 = 1$ and $f_1 = f_2 = 0$ to solve for $[h_0, h_1, h_2]$. This gives

$$h_0 = 1.111,\ \ h_1 = -0.1235,\ \ h_2 = 0.0137. \tag{9}$$

With these values $f_3 = h_2 g_1 = 0.0014$, and the output of the equaliser is

$$y_m = X_m \ + \ \underbrace{0.0014X_{m-3}}_{\text{residual interference}} \ + \ \underbrace{\sum_{i=0}^{2} h_i \tilde{n}_{m-i}}_{\text{residual noise}}.$$

## Question 2

(a) i) The TCP transmission rate is proportional to $\sqrt{\frac{1-q(t)}{q(t)}}$. Therefore, the factor by which the transmission rate decreases is [10%]

$$\frac{R_{0.1}}{R_{0.5}} = \frac{\sqrt{(1-0.1)/0.1}}{\sqrt{(1-0.05)/0.05}} = 0.6882.$$

ii) In the slow start phase, the window size is increased by 1 for each ACK received. It starts with $W = 1$, when the ack for the packet is received, $W = 2$. Then when the ack for these two packets are received, $W = 4$, and so on. [10%]

In the congestion avoidance phase, the window size increases by $1/W$ for each received ack. After an ack has been received for a window of $W$ packets, the window size is increased to $(W + 1)$.

iii) If the transmitter detects delay via three duplicate acks for the same packet, it retransmits the missing packet, and reduces the window size to $W/2$ (from $W$). On the other hand, if there is a timeout due to not receiving acks for any of the packets in the window, then the window size is set to $W = 1$. In both cases, the slow-start threshold `ssthresh` is set to $W/2$. [10%]

(b) i) We note that for $i \neq j$, the supports of $s_i(t)$ and $s_j(t)$ do not overlap. Therefore [10%]

$$\langle s_i(t),\ s_j(t)\rangle = \int_0^1 s_i(t)s_j(t)dt = \begin{cases} E, & i = j \\ 0, & i \neq j. \end{cases}$$

Therefore, an orthonormal basis is the set of functions $\{f_1(t), \ldots, f_M(t)\}$, where

$$f_i(t) = \frac{1}{\sqrt{E}}\, s_i(t) = \begin{cases} \sqrt{M}, & \text{for } \frac{(i-1)}{M} < t \leq \frac{i}{M} \\ 0, & \text{otherwise.} \end{cases}, \quad 1 \leq i \leq M.$$

The vectors are

$$\underline{s}_1 = [\sqrt{E}, 0, \ldots, 0], \ldots, \underline{s}_M = [0, \ldots, 0, \sqrt{E}],$$

with the only non-zero entry of $\underline{s}_i$ being in the $i$th position.

ii) The receiver first computes inner product of $y(t)$ with each of the basis functions to form $\underline{r} = [r_1, \ldots, r_M]$, where [20%]

$$r_1 = \int_{\mathbb{R}} y(t)f_1(t)dt, \quad \ldots, r_M = \int_{\mathbb{R}} y(t)f_M(t)dt.$$

If $x(t) = s_i(t)$, then the vector $\underline{r} = \underline{s}_i + \underline{n}$, where $\underline{n} = [n_1, \ldots, n_M]$ with

$$n_1 = \int_{\mathbb{R}} n(t)f_1(t)dt, \quad \ldots \quad, n_M = \int_{\mathbb{R}} n(t)f_M(t)dt.$$

Hence $\underline{r} = [n_1, \ldots, \sqrt{E}+n_i, \ldots, n_M]$, with the non-zero entry is in the $i$th position for $1 \leq i \leq M$. Further, $n_1, \ldots, n_M$ are i.i.d. $\sim \mathcal{N}(0, N_0/2)$. Hence, noting that the messages are a priori equally likely, the optimal decoding rule is decode message $\hat{m}$ where

$$\hat{m} = \arg\max_{1\leq i\leq M}\ r_i.$$

iii) By symmetry, we can calculate the probability of error assuming that message 1 (waveform $s_1(t)$) was transmitted. An error occurs if $r_1$ is *not* the maximum among $[r_1, \ldots, r_M]$. This implies [25%]

$$
\begin{aligned}
P_e &= P\left(\{r_1 \le r_2\} \cup \{r_1 \le r_3\} \cup \ldots \cup \{r_1 \le r_M\}\right) \\
&= P\left(\{\sqrt{E} + n_1 \le n_2\} \cup \{\sqrt{E} + n_1 \le n_3\} \ldots \cup \{\sqrt{E} + n_1 \le n_M\}\right) \\
&\le P\left(\{\sqrt{E} + n_1 \le n_2\}\right) + \ldots + P\left(\{\sqrt{E} + n_1 \le n_M\}\right) \\
&= (M-1) P\left(n_2 - n_1 \ge \sqrt{E}\right) \\
&\overset{(a)}{=} (M-1)\mathcal{Q}\left(\sqrt{\frac{E}{N_0}}\right) \overset{(b)}{=} (M-1)\mathcal{Q}\left(\sqrt{\frac{E_b \log M}{N_0}}\right)
\end{aligned}
$$

where step $(a)$ holds because because $(n_2 - n_1) \sim \mathcal{N}(0, N_0)$, and $(b)$ holds because each transmitted vector (message) corresponds to $\log M$ bits.

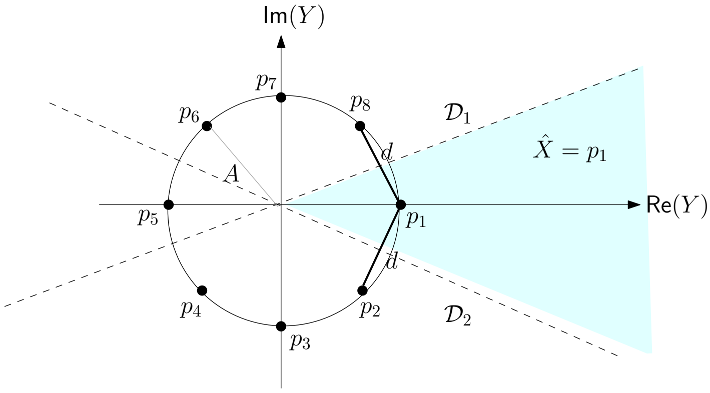
iv) Using $Q(x) \le e^{-x^2/2}$, the bound in part (iii) becomes [15%]

$$
P_e \le (M-1) e^{-\frac{E_b \log M}{2N_0}} \le e^{(\log_2 M)(\ln 2)} e^{-\frac{E_b \log_2 M}{2N_0}} = \exp\left(-\left(\frac{E_b}{N_0} - 2\ln 2\right)\log_2 M\right)
$$

Therefore, if $\frac{E_b}{N_0} > 2\ln 2$, then $P_e \to 0$ as $M \to \infty$. As $M$ increases, the number of orthogonal signalling dimensions (equal to $M$) increases, and hence the probability of error decreases. (An excellent answer will also mention that the decrease in probability of error comes at the expense of a decrease in bandwidth efficiency with growing $M$; but this is not expected as the question does not ask for this.)

## Question 3

(a) i) The constellation is shown in the figure, with the decision region for each point being a 45° wedge containing that point. [10%]




ii) Due to the circular symmetry of the constellation and the noise distribution, the probability of detection error is the same for each constellation symbol. Assuming that $p_1$ was the transmitted symbol, if $\mathcal{D}_1$ and $\mathcal{D}_2$ denote the two half-planes shown in the figure, the probability of error is

$$P_e = P(Y \in \mathcal{D}_1 \cup \mathcal{D}_2) \leq P(Y \in \mathcal{D}_1) + P(Y \in \mathcal{D}_2). \tag{10}$$

Let $N_1, N_2$ denote the components of the $N$ along the direction vector $(p_8 - p_1)$ and its orthogonal complement, respectively. Then due to the circular symmetry of the (complex) Gaussian distribution, the components $N_1, N_2$ are iid $\mathcal{N}(0, \frac{N_0}{2})$. Hence [25%]

$$P(Y \in \mathcal{D}_1) = P(N_1 > d/2) = \mathcal{Q}\left(\frac{d/2}{\sqrt{N_0/2}}\right),$$

where $d$ denotes the distance between adjacent constellation points. By the same argument, $P(Y \in \mathcal{D}_2) = \mathcal{Q}\left(\frac{d/2}{\sqrt{N_0/2}}\right)$. Substituting in (10), we obtain

$$P_e \leq 2\mathcal{Q}\left(\sqrt{\frac{d^2}{2N_0}}\right). \tag{11}$$

Finally, we note that $\frac{d}{2} = A\sin(\pi/8)$, and $A^2 = E_s = E_b \log_2 8 = 3E_b$. Substituting these in (11) yields the desired bound for the probability of decoding error of 8-ary PSK.

(b) i) Since the detector knows $s$, we now have an 8-PSK constellation with radius $\sqrt{s}A$. The energy per constellation symbol is $sA^2$, and hence the bound in part (a) applies with $E_b$ scaled by $s$: [10%]

$$P_{e,s} \leq 2\mathcal{Q}\left(\sqrt{\frac{6sE_b}{N_0}}\sin(\pi/8)\right).$$

ii) The probability of error bound averaged over all realizations of $s$ is [15%]

$$\begin{aligned}
P_e &= \mathbb{E}\left[2\mathcal{Q}\left(\sqrt{\frac{6sE_b}{N_0}}\sin(\pi/8)\right)\right] \\
&= 2\int_0^\infty e^{-s} \cdot \mathcal{Q}\left(\sqrt{\frac{6sE_b}{N_0}}\sin(\pi/8)\right)\, ds \\
&\leq \int_0^\infty e^{-s} \cdot e^{-\frac{3E_b}{N_0}\sin^2(\pi/8)s}\, ds = \frac{1}{1 + 3\sin^2(\pi/8)\frac{E_b}{N_0}}
\end{aligned}$$

where the inequality uses the bound $\mathcal{Q}(x) \leq \frac{1}{2}e^{-x^2/2}$, for $x \geq 0$.

iii) As $E_b/N_0$, increases, probability of error for the AWGN channel (in part (a) decreases exponentially with $E_b/N_0$), while the probability of error with fading decreases much more slowly — only as the inverse of $E_b/N_0$ as $E_b/N_0$ gets large. [10%]

Note: This is because there is a non-negligible probability of $s$ (and hence the effective snr) being very small, thus increasing the expected error probability. (This was not required to be mentioned in the answer.)

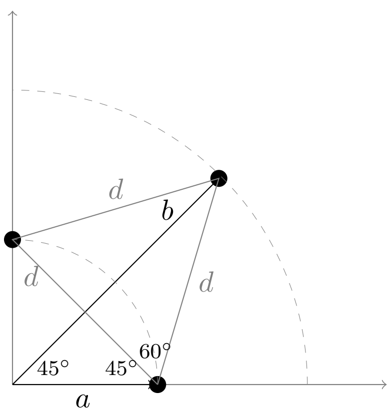
(c) From part (a), the distance between adjacent points of the constellation is $d = 2A\sin(\pi/8)$. [20%]

From the angles marked in the figure below we see that

$$a = \frac{d}{\sqrt{2}} = \sqrt{2}A\sin(\pi/8).$$

From the figure, we also deduce that

$$b = \frac{a}{\sqrt{2}} + d\sin 60^\circ = \frac{1+\sqrt{3}}{2}d = (1+\sqrt{3})A\sin(\pi/8).$$




(d) The average energy per symbol is [10%]

$$\frac{a^2+b^2}{2} = A^2(3+\sqrt{3})\sin^2(\pi/8) = 0.693A^2.$$

(The average symbol energy of the 8-PSK is $A^2$.)

## Question 4

(a) The table is given below. In each iteration $\omega_{ue}$ is the minimum-cost from each node $u$ to node $e$, and $n_{ue}$ is the next hop on the current min-cost path. [25%]

| Iteration | $(\omega_{ae}, n_{ae})$ | $(\omega_{be}, n_{be})$ | $(\omega_{ce}, n_{ce})$ | $(\omega_{de}, n_{de})$ |
|---|---|---|---|---|
| 0 | $(\infty, -)$ | $(7, e)$ | $(\infty, -)$ | $(2, e)$ |
| 1 | $(9, b)$ | $(5, d)$ | $(3, d)$ | $(2, e)$ |
| 2 | $(7, b)$ | $(4, c)$ | $(3, d)$ | $(2, e)$ |
| 3 | $(6, b)$ | $(4, c)$ | $(3, d)$ | $(2, e)$ |

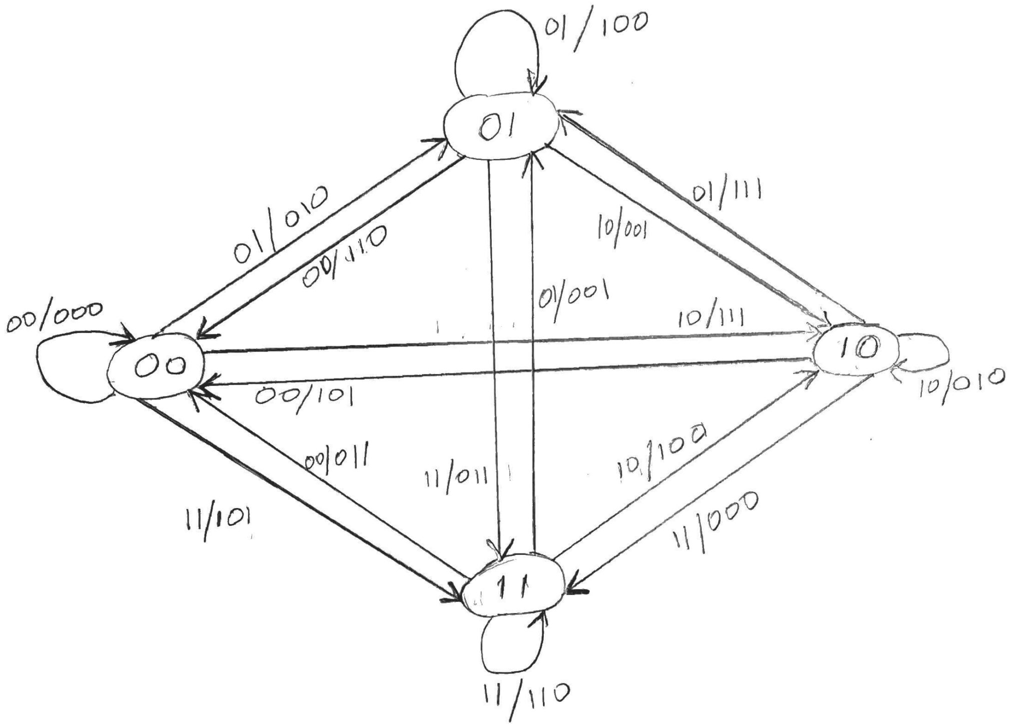
The minimum cost-paths are:

$$A - B - C - D - E: \text{ cost } 6$$
$$B - C - D - E: \text{ cost } 4$$
$$C - D - E: \text{ cost } 3$$
$$D - E: \text{ cost } 2$$

ii) To compute the shortest paths between each pair of nodes in the network, Djikstra's algorithm requires knowledge of the entire network topology and costs of all the paths in the network. On the other hand, in Bellman-Ford each node only requires knowledge of the costs to the nodes connected to it, and their current minimum costs to the destination node. Since each node uses only local information from its neighbours, it does not need to know the entire network topology. [15%]

(b) i) Since the encoder produces three coded bits for every two input bits, the rate of the code is 2/3.

ii) The state diagram of the code is shown in the figure below, with the label on each edge of the form 'input bits/output bits' corresponding to the transition: [25%]



ii) The free distance of a convolutional code is the minimum weight among codewords generated with the code starting and ending in the all-zero state. If the free distance is $d$, then any collection of $\lceil (d-1)/2 \rceil$ errors can be corrected, as long as these error bursts are not too close to one another. [15%]

iii) For convenience, label the states as follows: $00 \to a, \quad 01 \to b, \quad 10 \to c, \quad 11 \to d.$

To compute the free distance of the code, we only need to consider the codewords along three paths:

$$a-b-a, \qquad a-c-a, \qquad a-d-a.$$

This is because any longer paths starting and ending in the all-zero state *contain* the above paths, and hence cannot have weight smaller than the codewords along these paths. From the state diagram we see that: [15%]

The codeword along the path $a-b-a$ is 010 110 which has weight 3.

The codeword along the path $a-c-a$ is 111 101 which has weight 5.

The codeword along the path $a-d-a$ is 011 101 which has weight 4.

The free distance is therefore 3. Starting from the all-zero state, the input sequence 01 00 produces the output sequence 010 110, and returns to the all-zero state.